

Nemesch, Levin; Ruzika, Stefan; Thielen, Clemens; Wittmann, Alina

Article — Published Version

A survey of exact and approximation algorithms for linear-parametric optimization problems

Journal of Global Optimization

Provided in Cooperation with:

Springer Nature

Suggested Citation: Nemesch, Levin; Ruzika, Stefan; Thielen, Clemens; Wittmann, Alina (2025) : A survey of exact and approximation algorithms for linear-parametric optimization problems, Journal of Global Optimization, ISSN 1573-2916, Springer US, New York, NY, Vol. 93, Iss. 1, pp. 299-333, <https://doi.org/10.1007/s10898-025-01512-6>

This Version is available at:

<https://hdl.handle.net/10419/330542>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by/4.0/>



A survey of exact and approximation algorithms for linear-parametric optimization problems

Levin Nemesch¹ · Stefan Ruzika¹ · Clemens Thielen² · Alina Wittmann² 

Received: 16 January 2025 / Accepted: 12 June 2025 / Published online: 28 June 2025
© The Author(s) 2025

Abstract

Linear-parametric optimization, where multiple objectives are combined into a single objective using linear combinations with parameters as coefficients, has numerous links to other fields in optimization and a wide range of application areas. In this survey, we provide a comprehensive overview of structural results and algorithmic strategies for solving linear-parametric optimization problems exactly and approximately. Transferring concepts from related areas such as multi-objective optimization provides further relevant results. The survey consists of two parts: First, we list strategies that work in a general fashion and do not rely on specific problem structures. Second, we look at well-studied parametric optimization problems and cover both important theoretical results and specialized algorithmic approaches for these problems. Among these problems are parametric variants of shortest path problems, minimum cost flow and maximum flow problems, spanning tree problems, the knapsack problem, and matching problems. Overall, we cover the results from 128 publications (and refer to 35 supplemental works) published between 1963 and 2024.

Keywords Parametric optimization · Parametric programming · Approximation algorithms · Combinatorial optimization · Survey

1 Introduction

Linear-parametric optimization has been an important research area for more than 60 years with hundreds of articles published on the topic and a wide range of applications.

✉ Alina Wittmann
alina.wittmann@tum.de

Levin Nemesch
l.nemesch@math.rptu.de

Stefan Ruzika
stefan.ruzika@math.rptu.de

Clemens Thielen
clemens.thielen@tum.de

¹ Department of Mathematics, RPTU Kaiserslautern-Landau, Paul-Ehrlich-Str. 31, 67663, Kaiserslautern, Germany

² Campus Straubing for Biotechnology and Sustainability, Professorship of Optimization and Sustainable Decision Making, Technical University of Munich, Am Essigberg 3, 94315, Straubing, Germany

In a linear-parametric optimization problem, different objectives are combined into a single objective by a linear combination with parameters as coefficients. The goal is to find an optimal solution set, that is, a set containing an optimal solution for every possible combination of parameter values. This formulation naturally arises in various real-world applications where parameters change over time or across different scenarios. In recent decades, the theory of parametric optimization has expanded significantly, along with its diverse applications in fields such as biomedical systems, power electronics, fleet planning, aeronautics, model predictive control, and process synthesis under uncertainty [1–3]. Furthermore, parametric optimization has many connections to other research areas, including computational geometry and various areas of optimization, such as multi-objective optimization, Lagrangian relaxation, robust optimization, sensitivity analysis, or multi-level optimization [4–9].

By fixing the parameters of a parametric optimization problem, it becomes a conventional non-parametric optimization problem. The parametric problem then is at least as hard as the resulting non-parametric problem. But even if the non-parametric problem can be solved in polynomial time, the parametric problem itself might be intractable. The reason for this is that, for many parametric problems, optimal solution sets can have a cardinality that is exponential in the encoding length of the problem input. This includes, among others, the parametric variants of shortest path problems [10], the assignment problem [11], and the minimum cost flow problem [12]. Still, some problems such as the parametric minimum spanning tree problem remain tractable [13].

Both the facts that a parametric problem can be intractable and that it can be computationally hard for a fixed parameter value motivate the notion of approximation. Similarly to approximation in conventional optimization, approximation algorithms for parametric problems allow finding solution sets in polynomial time while ensuring a strict guarantee on the (worst-case) quality of the solutions [14–18].

This survey provides a comprehensive overview of theoretical developments, algorithmic strategies, and other recent advances in linear-parametric optimization, with a particular focus on some selected combinatorial optimization problems. For parametric mixed integer linear programming, extensive surveys are provided in [1, 2, 19–21]. We refer to these surveys for an in-depth introduction. Our focus is on general, problem-agnostic approaches, approximation algorithms, and results for specific parametric problems.

1.1 Scope and outline

This survey covers structural and algorithmic results for (multi-)parametric optimization problems. In particular, cardinality results concerning optimal solution sets as well as exact solution methods and approximation algorithms for linear-parametric problems are addressed.

Parametric optimization is related to many other areas in optimization. Some of these relationships are briefly highlighted in Section 2. In Section 3, we discuss general methods and results that apply to larger classes of parametric optimization problems. We take advantage of the fact that some definitions and concepts in parametric optimization can also be found identically in multi-objective optimization. Thus, we adopt some results from multi-objective optimization. In Section 4, we focus on methods and results for parametric variants of specific problems, in particular, well-known combinatorial optimization problems. The selected problems are among the most-studied problems in the context of parametric optimization throughout peer-reviewed journals and conference proceedings in the English

language. Table 2 in Appendix A provides an overview of the problem classes considered in this survey.

1.2 Preliminaries and definitions

We distinguish between the natural numbers \mathbb{N} starting with 1 and \mathbb{N}_0 starting with 0. Given $p, n \in \mathbb{N}$, let $[p] := \{1, \dots, p\}$ be the set of natural numbers up to $p \in \mathbb{N}$. We denote the nonnegative orthant by $\mathbb{R}_{\geq 0}^p := \{y \in \mathbb{R}^p : y \geq 0\}$, where $0 \in \mathbb{R}^p$ is the p -dimensional zero vector and \geq is the weak component-wise order defined by

$$y \geq y' \text{ if and only if } y_i \geq y'_i \quad \forall i \in [p].$$

Given a function $f : \mathbb{R}^p \rightarrow \mathbb{R}^n$ and a set $Z \subseteq \mathbb{R}^p$, we denote the image of Z under f by $f(Z) := \{f(z) : z \in Z\} \subseteq \mathbb{R}^n$.

For running time specifications, we use the well-known Landau notation. We also use $\tilde{\mathcal{O}}(\dots)$ (read soft- \mathcal{O}), an extension to the Landau notation that hides polylogarithmic factors, e.g., $\tilde{\mathcal{O}}(g(n))$ for $\mathcal{O}(g(n) \log^k(n))$ for some $k \in \mathbb{N}$. For a problem P , T_P denotes the running time of an oracle that solves P . A polynomial in n is referred to as $\text{poly}(n)$. An algorithm is *(input-)polynomial* if, for an input of size I , it has a running time in $\mathcal{O}(\text{poly}(I))$. If no polynomial time algorithm exists for a problem, it is called *intractable*. An algorithm is *output-polynomial* if, for an input of size I and an output of size O , it has a running time in $\mathcal{O}(\text{poly}(I, O))$.

The following definitions introduce general (multi-)parametric minimization problems [17, 18]. Note that, for the sake of simplicity, we consider minimization problems, but all concepts can be defined analogously for maximization problems.

Definition 1 (Parameter Set) A *parameter set* $\Lambda \subseteq \mathbb{R}^p$ is a set of the form

$$\Lambda := \{\lambda \in \mathbb{R}^p : \lambda_{\min} \leq \lambda \leq \lambda_{\max}\}$$

for a given $\lambda_{\min} \in (\mathbb{R} \cup \{-\infty\})^p$ and a given $\lambda_{\max} \in (\mathbb{R} \cup \{\infty\})^p$.

We remark that parametric optimization problems can easily be defined for more complex parameter sets, e.g., for polyhedral or non-convex parameter sets. Almost all parameter sets used in the literature, however, are covered by Definition 1.

Definition 2 ((Multi-)Parametric Minimization Problem) Let $p \in \mathbb{N}$. A *linear (p-)parametric minimization problem* $\Pi = (X, f, \Lambda)$ is a triple consisting of

- (a) a nonempty set $X \subseteq \mathbb{R}^n$ of feasible solutions for some $n \in \mathbb{N}$,
- (b) a function $f : X \rightarrow \mathbb{R}^{p+1}$, $x \mapsto f(x) = (f_0(x), f_1(x), \dots, f_p(x))^T$, and
- (c) a *parameter set* $\Lambda \in \mathbb{R}^p$.

A solution of Π is a set $S \subseteq X$ that contains, for every $\lambda \in \Lambda$, an optimal solution of the problem

$$\min_{x \in X} F_\lambda(x) := \min_{x \in X} f_0(x) + \sum_{i=1}^p \lambda_i f_i(x). \quad (\Pi(\lambda))$$

For a fixed parameter value $\lambda \in \Lambda$, the problem $\Pi(\lambda)$ is a conventional (non-parametric) minimization problem. The *optimal cost curve* $F^* : \Lambda \rightarrow \mathbb{R} \cup \{-\infty\}$, $\lambda \mapsto F^*(\lambda) := \inf_{x \in X} F_\lambda(x)$ assigns to every $\lambda \in \Lambda$ the optimal objective value of $\Pi(\lambda)$, or $-\infty$ if $\Pi(\lambda)$ is unbounded. The set $F^*(\Lambda)$ is called the *lower envelope* of Π .

With ALG, we denote an oracle for a problem Π that takes as input any $\lambda \in \Lambda$ and computes an optimal solution of $\Pi(\lambda)$.

For an optimal solution set S , we do not only expect it to contain an optimal solution for every $\lambda \in \Lambda$, but to also have minimal cardinality among all such sets. With some slight abuse of notation, B denotes the cardinality $|S|$ if it is finite or ∞ if it is infinite.

The *critical region* $\Lambda(x) \subseteq \Lambda$ of a solution $x \in X$ is the set of all parameters for which x is an optimal solution of $\Pi(\lambda)$. In the case of a single-parametric problem, the critical region of each solution in an optimal solution set is an interval in \mathbb{R} . The point where two adjacent intervals intersect is called a *breakpoint*.

As the cardinality of an optimal solution set can be superpolynomially large in the worst case, even if $p = 1$, approximation provides a good concept to substantially reduce the number of required solutions while still obtaining provable solution quality.

Definition 3 For an approximation factor $\alpha \geq 1$ and a fixed parameter vector $\lambda \in \Lambda$, a feasible solution $x \in X$ is called α -approximate for the problem $\Pi(\lambda)$ if $F_\lambda(x) \leq \alpha \cdot F_\lambda(x')$ for all $x' \in X$.

A necessary condition for the existence of an α -approximate solution of $\Pi(\lambda)$ is that the optimal objective value of $\Pi(\lambda)$ is greater than or equal to zero. Therefore, it is assumed that $F^*(\lambda) \geq 0$ for all $\lambda \in \Lambda$ whenever parametric approximation algorithms are considered.

Usually, the goal in parametric approximation is to find a set of solutions that contains an α -approximate solution for each non-parametric problem $\Pi(\lambda)$:

Definition 4 (Approximation set) For an approximation factor $\alpha \geq 1$, a finite set $S \subseteq X$ is called an α -approximation set to Π if it contains an α -approximate solution $x \in S$ of $\Pi(\lambda)$ for every $\lambda \in \Lambda$. An α -approximation set is *minimal* if removing any solution would no longer provide an α -approximation set.

An algorithm A that computes an α -approximation set for an instance Π in time polynomially bounded in the instance size is called an α -approximation algorithm.

A *polynomial-time approximation scheme* (PTAS) is a family of algorithms $(A_\varepsilon)_\varepsilon$ such that, for any $\varepsilon > 0$, the algorithm A_ε is a $(1 + \varepsilon)$ -approximation algorithm. A PTAS $(A_\varepsilon)_\varepsilon$ is a *fully polynomial-time approximation scheme* (FPTAS) if the running time of A_ε is also polynomial in $1/\varepsilon$.

Similar to the oracle ALG, an *approximation oracle* ALG_β for a $\beta \geq 1$ takes as input any $\lambda \in \Lambda$ with $F^*(\lambda) \geq 0$ and returns a β -approximate solution of the problem $\Pi(\lambda)$.

2 Related Areas

We relate parametric optimization to some other selected areas of mathematical optimization. The purpose is twofold: We demonstrate the impact of parametric optimization and, vice versa, we illustrate how parametric optimization may benefit from progress made in related fields of research.

In Lagrangian relaxation, constraints are moved into the objective function together with a Lagrangian multiplier such that any violation of the constraints is penalized [22].

$$\begin{array}{ll} \min_x & f_0(x) \\ \text{s.t.} & f_1(x) \leq 0 \\ & g(x) \leq 0 \end{array} \quad \rightarrow \quad \left\{ \begin{array}{ll} \min_x & f_0(x) + \lambda \cdot f_1(x) \\ \text{s.t.} & g(x) \leq 0 \end{array} \right\}_{\lambda \in \Lambda}$$

The degree of penalization depends on the Lagrangian multiplier which influences the quality of the dual bound obtained. Lagrangian relaxation and solving the Lagrangian dual problem can be viewed as a parametric optimization problem (see [5, 23, 24]).

In many domains such as engineering, computer science, or finance, data of related to the objective function and / or the constraints of optimization problems is often uncertain. Robust optimization aims to find solutions that are robust with respect to uncertainty described by realizations of parameter values within a given set of scenarios. Solution approaches for robust optimization problems are developed with the help of parametric optimization by considering the uncertainty set as the parameter set, which means that solving the parametric problem provides a solution for every possible scenario in the uncertainty set [6, 25–27].

In the field of sensitivity analysis, the behavior of primal solutions is examined when there is perturbation or uncertainty in the input parameters or the structure of an optimization problem. This means that results and methods for parametric optimization problems can also be found in the literature that is concerned with sensitivity analysis for combinatorial optimization problems (see [28, 29]). An overview on the historical development and literature overview of sensitivity analysis and parametric optimization can be found in [7].

A multi-level optimization problem consists of several optimization problems embedded in each other, where the constraints of the problem on one level are implicitly determined by the problems on the lower levels. Multi-parametric optimization is used as a tool for solving multi-level optimization problems (see [1, 8, 30–32]). The idea is to consider the lowest-level problem as a multi-parametric problem with the higher-level decision variables as parameters. Then, this parametric problem is solved, and each critical region defines a new multi-level problem having one level less. Iteratively, this process is continued until one single level remains [8].

Tokuyama [9] compares the trajectories of all weight functions of a parametric optimization problem in a plane with line arrangements in computational geometry. Apart from line arrangements, lower envelopes of line segments and k -sets are also connected to parametric optimization [33].

2.1 Multi-Objective optimization

An area with particularly close connections to parametric optimization is *multi-objective optimization*. We only outline some key concepts here. For an in-depth introduction to multi-objective optimization, we refer to Ehrgott [34].

Given a set $X \in \mathbb{R}^n$ of feasible solutions and a (vector-valued) objective function $f = (f_1, \dots, f_d): X \rightarrow \mathbb{R}^d$ for $d \in \mathbb{N}_{>1}$, a *d-objective optimization problem* is of the form:

$$\min_{x \in X} (f_1(x), \dots, f_d(x)) \quad (\Pi^{\text{MO}})$$

Since solutions optimizing f_1, \dots, f_d simultaneously do typically not exist for a d -objective problem, one is usually interested in so-called *efficient solutions* [34]. Here, a feasible solution $x \in X$ is called *efficient* (or *Pareto optimal*) if there is no other feasible solution $x' \in X$ such that $f(x') \leq f(x)$ and $f(x') \neq f(x)$. The image $f(x)$ of an efficient solution x under f is called a *non-dominated* image. The goal in a multi-objective optimization problem then typically consists of determining the set of all non-dominated images (the *non-dominated set*) and one corresponding efficient solution for each of these non-dominated images.

A significant part of the multi-objective optimization literature, however, is mainly interested in a subset of the non-dominated set, namely in the set of non-dominated extreme points of the convex hull $\text{conv}(Y)$ of the image set $Y = f(X)$. This is where the close rela-

tion between multi-objective and parametric optimization arises. The set of non-dominated extreme points of $f(X)$ can also be characterized as an optimal solution set of the parametric problem $\min_{x \in X} \lambda^\top f(x)$ with the parameter set $\Lambda = \mathbb{R}_{\geq 0}^d$ [34]. In the multi-objective literature, this parameter set is called the *weight set*. Hence, solving this parametric problem and finding the non-dominated extreme points are equivalent problems.

This relation led to the development of some similar approaches in the parametric and the multi-objective optimization literature. A classic example for this is the famous technique known as the *Eisner-Severance method* in parametric optimization and as the *dichotomic approach* in multi-objective optimization (see Section 3.1). In Section 3.3, we provide an overview of some algorithmic strategies that were independently developed in both areas and whose connection has, to the best of our knowledge, not previously been outlined in the literature.

The similarities between parametric and multi-objective optimization also allow approximation algorithms from the multi-objective setting to be used in the parametric setting. As long as $\Lambda = \mathbb{R}_{\geq 0}^d$, every so-called α -convex Pareto set for the corresponding multi-objective problem is an α -approximation set of the parametric problem, and vice versa [4]. Under the same condition, every so-called α -approximate Pareto set (a stricter notion of approximation in the multi-objective setting) is also a parametric α -approximation set, but it might contain solutions that are redundant for the latter. These can be filtered out by convex hull algorithms [15]. For a comprehensive overview on multi-objective approximation, we refer to the recent survey in [35].

3 General Methods

In this section, we present general exact and approximate methods that are applicable for a broad class of parametric optimization problems, independent of specific problem structures. The Eisner-Severance method is discussed in Section 3.1, Gusfield's algorithm using Megiddo's method in Section 3.2, the connection between parametric and multi-objective optimization in Section 3.3, parametric approximation in Section 3.4 and results for parametric mixed integer linear programs are reviewed in Section 3.5.

3.1 The Eisner-Severance method

A widely-used algorithm is the *Eisner-Severance method* [36], which has also been introduced by the multi-objective literature under a different name. Eisner and Severance aimed at record-segmentation problems in shared databases, but their method can be used as a general algorithm to solve single-parametric optimization problems. The precise origin of the method is unclear, as Nauss [37] independently developed a similar approach for Knapsack problems in 1975. Radke [38] and Geoffrion and Nauss [19] both generalize Nauss' algorithm to more generic problems and provide proofs of correctness and running time. Gusfield [28] combines both branches of the literature and states a refined proof of correctness and running time.

The Eisner-Severance method works on single-parametric problems with parameter sets of the form $[\lambda_{\min}, \lambda_{\max}]$. It requires access to an oracle ALG. The method is illustrated in Figure 1(a) from the parametric perspective and in Figure 1(b) from the multi-objective perspective. After initializing $\{\lambda_{\min}, \lambda_{\max}\}$, two solutions $x_1 := \text{ALG}(\lambda_{\min})$ and $x_2 := \text{ALG}(\lambda_{\max})$ are computed. Then, the parameter value λ^* is computed for which the objective function value of both solutions is equal, i.e., the point where the functions $F_{\lambda^*}(x_1)$ and $F_{\lambda^*}(x_2)$

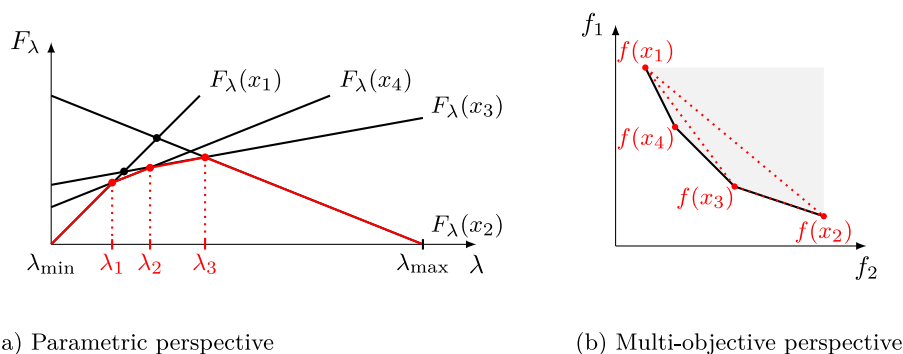


Fig. 1 Example showing two different perspectives on the Eisner-Severance method [36], returning the optimal cost curve in the parametric perspective in (a) and the non-dominated extreme points in the multi-objective perspective in (b)

intersect in Figure 1(a) or the normal of the connecting line between $f(x_1)$ and $f(x_2)$ in Figure 1(b). If $F_{\lambda^*}(x_1)$ is the optimal objective value at λ^* , then λ^* is a breakpoint, x_1 is optimal in $[\lambda_{\min}, \lambda^*]$ and x_2 is optimal in $[\lambda^*, \lambda_{\max}]$. Thus, x_1 and x_2 together are an optimal solution set in the interval $[\lambda_{\min}, \lambda_{\max}]$. Otherwise, the above procedure is applied recursively to the two sub-intervals $[\lambda_{\min}, \lambda^*]$ and $[\lambda^*, \lambda_{\max}]$.

Since all other operations can be performed in $\mathcal{O}(1)$, the number of calls to ALG determines the running time of the Eisner-Severance method. It is called at least $2B - 1$ and at most $3B - 2$ times (cf. [39]). Under the additional assumption that ALG only outputs solutions that correspond to line segments of the optimal cost curve, exactly $2B - 1$ calls to ALG are needed [28, 40].

An approximation algorithm for single-parametric problems based on the Eisner-Severance method is developed by Bazgan et al. [17]. If a β -approximation oracle ALG_β for $\Pi(\lambda)$ and all $\lambda \in \Lambda$ is available, their algorithm is able to compute a $(1 + \varepsilon)\beta$ -approximation set for any $\varepsilon > 0$. Both the cardinality of the approximation set and the number of calls to ALG_β are bounded by a polynomial in the size of the input and $1/\varepsilon$. Figure 2 illustrates this approach.

A strategy that works functionally identical to the Eisner-Severance method was independently developed in multi-objective optimization to identify non-dominated extreme points of bi-objective optimization problems. There, it is known as the *dichotomic approach* and first appears in [40–42]. An approximate variant of this algorithm also exists. It is known as the *Sandwich Algorithm with Underlying Chord Partition Rule* [43, 44] or the *Chord Algorithm* [45, 46], and is illustrated in Figure 3.

3.2 Gusfield's algorithm and Megiddo's method

Another general algorithm for single-parametric optimization problems is described by Gusfield [47]. It is based on a well-known technique from single-parametric search, called *Megiddo's method*. First described by Megiddo [48] for parametric search in combinatorial optimization problems, Megiddo's method is developed further in [49–52], adapted for non-linear problems in [53, 54], and turned into an approximation algorithm in [55, 56].

The goal of Megiddo's method is to find a parameter value λ^* for which some condition holds – usually a parameter value where F^* attains its global optimum. As long as a

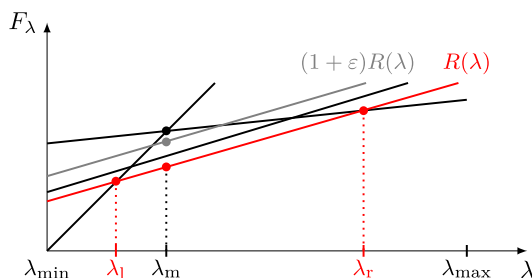


Fig. 2 Approximate variant of the Eisner-Severance method from [17]. The algorithm starts with a parameter interval $([\lambda_l, \lambda_r], \text{ALG}_\beta(\lambda_l), \text{ALG}_\beta(\lambda_r))$. First, it is checked whether the optimal / approximate solution $x_l := \text{ALG}_\beta(\lambda_l)$ for λ_l or $x_r := \text{ALG}_\beta(\lambda_r)$ for λ_r is α -approximate for the whole interval $[\lambda_l, \lambda_r]$ by examining the two conditions $F_{\lambda_r}(x_l) \leq F_{\lambda_r}(x_r)$ and $F_{\lambda_l}(x_r) \leq F_{\lambda_l}(x_l)$. Otherwise, it is checked whether the intersection point (λ_m, y_m) of $F_{\lambda_l}(x_l)$ and $F_{\lambda_r}(x_r)$ lies below the red connecting line $R(\lambda)$ shifted by a factor of $1 + \epsilon$, i.e., below the gray line $(1 + \epsilon)R(\lambda)$. Otherwise, the algorithm is recursively called with $([\lambda_l, \lambda_m], x_l, \text{ALG}_\beta(\lambda_m))$ and $([\lambda_m, \lambda_r], \text{ALG}_\beta(\lambda_m), x_r)$

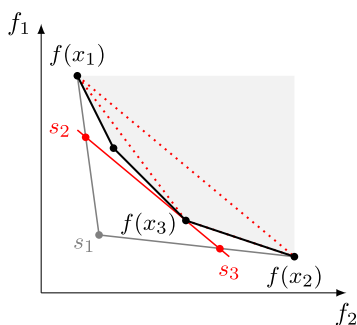


Fig. 3 Chord Algorithm. Again, the aim is to find a solution that is optimal with respect to the normal of the connecting line between $f(x_1)$ and $f(x_2)$. Let the two gray lines be the connecting lines from the previous iterations shifted to the optimal points. In contrast to the exact Eisner-Severance method, the recursion already terminates if the connecting line between $f(x_1)$ and $f(x_2)$ is not more than the approximation factor away from the intersection point s_1 or the new optimal / approximate image $f(x_3)$

polynomial-time algorithm for $\Pi(\lambda)$ is available for every $\lambda \in \Lambda$, Megiddo's method also runs in polynomial time. The basic idea is to simulate the algorithm for $\Pi(\lambda^*)$ without actually knowing the value of λ^* . Whenever the simulated algorithm encounters a conditional branching, Megiddo's method needs to detect which branch the algorithm would take for the (still unknown) input λ^* .

Any conditional branching that is encountered is a less-or-equal comparison between two linear functions of λ (or can be decided trivially). The result of the comparison is determined by the parameter value of λ^* . However, it can also be decided by solving $\Pi(\lambda)$ for the parameter value for which the two functions intersect. Solving $\Pi(\lambda)$ can be done with the same algorithm that is simulated for λ^* . When the simulated algorithm terminates, the value of λ^* and a solution of $\Pi(\lambda^*)$ are known.

Gusfield's algorithm transforms this parametric search technique into an algorithm that computes an optimal solution set. It uses Megiddo's method to find an initial breakpoint, and then to iteratively identify subsequent breakpoints with corresponding solutions. The running time of Gusfield's algorithm is in $\mathcal{O}(B \cdot T_\Pi^2)$ [47].

3.3 Minimization diagram and lower image

Next, we look at some general strategies that are applicable to multi-parametric mixed integer linear programs or multi-parametric combinatorial problems. We also include some algorithms from multi-objective optimization that, fundamentally, take the parametric perspective to find non-dominated extreme points. To keep the notation simple, we directly transfer all results from the multi-objective literature into the parametric setting.

3.3.1 Decomposition of the parameter set and minimization diagrams

For every $x \in X$ where X is described by a system of mixed integer linear constraints, the critical region $\Lambda(x)$ is a convex polyhedron. Several approaches decompose the parameter set Λ into disjoint p -dimensional critical regions and find one solution per critical region. Such a subdivision is called a *minimization diagram* [57]. An example of a minimization diagram can be found in Figure 4(a). In multi-objective optimization, similar approaches have been developed. There, the term *weight set decomposition* [58] is used and the weight set is decomposed. A formal description of the theoretical properties of minimization diagrams is given by so-called *Davenport-Shinzel sequences*, see [57, 59]. Another rigorous analysis can be found in the multi-objective literature in [58].

Since nearly all available literature is concerned with the 2-parametric case (e.g., [47, 58, 60–63]), we assume that $p = 2$ for the rest of this section, unless stated otherwise. If a problem uses more than two parameters, strategies like the ones outlined in Section 3.3.2 are often applied.

Deciding if two critical regions share an edge and, if so, finding the vertices of this edge can be done by solving a single-parametric problem with any algorithm (e.g., the methods from Sections 3.1 and 3.2). This problem can be seen as intersecting the parameter set Λ with a straight line. Hence, procedures that investigate potential edges of critical regions are called *line searches*. As notational shorthand throughout this section, we denote the number of vertices in a minimization diagram by V and let E denote the number of edges in it.

Nearly all algorithms that compute minimization diagrams follow a similar pattern: An initial solution (or a set of initial solutions) is computed, and its critical region is constructed by a sequence of line searches. During these line searches, new solutions are discovered, for which the process is repeated until all critical regions have been found.

The first such algorithm in parametric literature is presented by Gusfield [47]. Here, critical regions are computed by simulating ray shooting with line searches. Rays are shot from the interior of a critical region to determine its edges. The running time bound of the procedure is stated as $\mathcal{O}(E \cdot T_{\Pi}^2)$ ¹.

Fernández-Baca [60] develops a sweep-line approach: Edges of the minimization diagram are traced via line searches, and its vertices are enumerated in ascending order of their λ_1 value. The running time is in $\mathcal{O}(E \cdot T_{\Pi}^2 + E \log E)$. A generalization for the 3-parametric case is outlined shortly.

Waterman et al. [61] and Zimmer and Lengauer [62] develop algorithms for genetic alignment problems, which can also be used as general strategies for finding minimization diagrams. The first, given by Waterman et al. [61], operates by tracing the boundary of each critical region separately. No running time analysis is given for this algorithm. In the algorithm from Zimmer and Lengauer [62], a graph is used to encode adjacency relations

¹ Fernández-Baca [60] claims that the running time analysis by Gusfield [47] contains an error, but no details are given on the specific nature of this error.

of critical regions in the minimization diagram. This is used to systematically check and identify potential edges. The running time is in $\mathcal{O}(B^4 + (B + V)T_{\Pi})$, and a computational study in [62] shows that this approach outperforms the algorithm from Waterman et al. [61] on genetic alignment problems.

In the field of multi-objective optimization, the first algorithm that computes a minimization diagram was developed by Benson and Sun [64], but can only be used for multi-objective linear programming problems. The first two general algorithms were published by Przybylski et al. [58]. One of them is called the *brute force algorithm* and works almost identically to the algorithm of Zimmer and Lengauer [62]. The other algorithm tries to improve the practical running time of the brute force algorithm by eliminating redundant line searches. A computational study shows no significant advantage for either of the two variants. Another algorithm is presented by Halffmann [63]. It operates by constructing inner approximations for the critical regions and improving them iteratively through line searches. The running time of this algorithm is in $\mathcal{O}(B^2 T_{\Pi})$, and a computational study shows a similar performance to the algorithm of Przybylski et al. [58] in a computational study. Some further improvements are developed in [65].

Alves and Costa [66] present an algorithm for computing a subset of a critical region. The approach can be used to compute the entire minimization diagram, however, no running time bound is stated in [66].

3.3.2 Lower image computation

An approach that is closely related to parameter set decomposition is the construction of the so-called *lower image* \mathcal{L} . While the concept appears earlier in the parametric literature², the term “lower image” stems from multi-objective optimization [68] and is the first name explicitly given to the polyhedron \mathcal{L} . Again, we directly take the parametric perspective when describing results from multi-objective optimization.

Definition 5 (Heyde and Löhne [68]) The *lower image* \mathcal{L} is the set

$$\mathcal{L} := \{(\lambda, z) : \lambda \in \Lambda, z \leq F^*(\lambda)\}.$$

The lower image can equivalently be described by the solution set: For every solution $x \in X$, the set $H(x) = \{(\lambda, z) \in \mathbb{R}^p \times \mathbb{R} : z \leq F_{\lambda}(x)\}$ describes a halfspace. The intersection of $\Lambda \times \mathbb{R}$ and all halfspaces $H(x)$ for $x \in X$ is exactly the lower image [68]. However, the halfspaces from any optimal solution set are sufficient to obtain a minimal description of \mathcal{L} . This fact is used by several algorithmic approaches that construct \mathcal{L} using techniques from computational geometry to obtain an optimal solution set.

The concept of the minimization diagrams and the lower image are closely related. A minimization diagram can be obtained by projecting \mathcal{L} onto the λ plane, and \mathcal{L} can be constructed by “lifting” each critical region in the diagram with its objective value. This is illustrated in Figure 4.

Approaches that construct \mathcal{L} all follow a similar pattern: Solutions are found iteratively, and, in each iteration, the currently known solutions are used to construct a polyhedral approximation of \mathcal{L} . From this approximation, parameter values are obtained, and the corresponding non-parametric problem is solved for these values to either find new solutions or determine that an optimal solution set has been found. The first such algorithm is limited to the 2-parametric case and is described by Fernández-Baca and Srinivasan [67]. Here, the

² To the best of our knowledge, the first mention is by Fernández-Baca and Srinivasan [67].

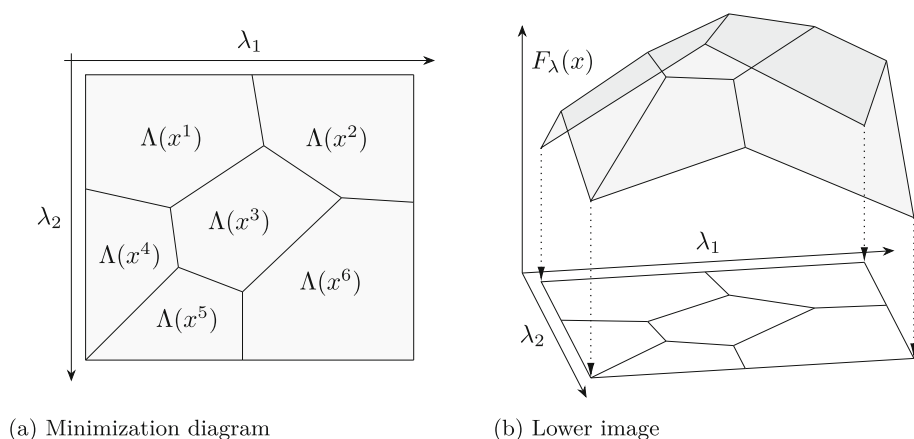


Fig. 4 The minimization diagram and the lower image for an instance with $B = 6$. The shadow of the lower image is exactly the minimization diagram

polyhedral approximation is maintained by a graph structure. The algorithm is generalized by Fernández-Baca and Srinivasan [69] to the general p -parametric case³ by using a procedure described by Dobkin et al. [70] for the polyhedral approximation.

In the multi-objective optimization literature, the so-called *dual Benson algorithm* is used to construct \mathcal{L} . First presented by Ehrgott et al. [71], it was further developed in [72, 73]. Originally designed for multi-objective linear programs, it was subsequently generalized to apply to combinatorial problems [74], integer linear programs [75], and mixed integer linear programs [76]. In the dual Benson algorithm, the well-known *vertex enumeration problem* [77] is used for the polyhedral approximation. Bökler and Mutzel [74] show that the dual Benson algorithm runs in output-polynomial time as long as $\Pi(\lambda)$ can be solved in polynomial time for every $\lambda \in \Lambda$. A number of solvers based on the dual Benson algorithm (or similar strategies) have been developed: *bensolve* [73] and *inner* [39] for linear programs, *PolySCIP* [75] for integer linear programs, and *PaMILO* [76] for mixed integer linear programs.

In multi-objective optimization, there are also a number of algorithms that compute the so-called *Edgeworth-Pareto hull* (often also referred to as the *upper image*). The Edgeworth-Pareto hull is the polyhedron $\text{conv}(Y) + \mathbb{R}_{\geq 0}^d$, which is dual to \mathcal{L} . This duality relation is described in detail by Heyde and Löhne [68] for linear programming, but also applies to general mixed integer linear programming [78]. Because of this duality relation, every algorithm that computes the Edgeworth-Pareto hull can also be used to compute \mathcal{L} . Such algorithms can be found in [39, 78–81].

3.4 General approximation algorithms for parametric optimization problems

General approximation algorithms for parametric problems usually work under rather mild assumptions (see, for example, [17, 18]):

1. For every $i \in [p]$, the function f_i is non-negative and f is computable in polynomial time. Furthermore, $F^*(\lambda) \geq 0 \forall \lambda \in \Lambda$.

³ At this point in time, the algorithm Fernández-Baca and Srinivasan [69] describe is already considered as “part of the folklore”. But to the best of our knowledge, no formal description of the algorithm appears in literature before [69]

2. An approximation oracle ALG_β with polynomial running time is available.
3. Often, the structure of Λ is further constrained, e.g., by setting $\Lambda = \mathbb{R}_{\geq 0}^p$ [4] or by bounding each parameter value from below [17, 18].

There exist only a few general approximation algorithms for parametric optimization problems. For the single-parametric case, the adaption of the Eisner-Severance method by Bazgan et al. [17] is available (see Section 3.1).

For multi-parametric problems, an early approach is described by Katoh and Ibaraki [82], but it is only used as part of a larger algorithm there. A dedicated approximation algorithm for multi-parametric problems is developed by Helfrich et al. [18]. The approaches of both Katoh and Ibaraki [82] and Helfrich et al. [18] work by constructing a grid over the parameter set. For each vertex of this grid, the non-parametric problem is solved using ALG_β . The structure of the grid then ensures that the returned solution is $(\alpha \cdot \beta)$ -approximate for some $\alpha > 1$ for parameter values in a small neighborhood of this vertex. Furthermore, the union of these neighborhoods covers the entire parameter set, so the set of all solutions returned by the oracle calls at the vertices is an $(\alpha \cdot \beta)$ -approximation set. If the oracle is exact or an (F)PTAS, the parametric approximation algorithm itself is an (F)PTAS. When comparing the algorithm of Katoh and Ibaraki [82] to that of Helfrich et al. [4], the latter needs fewer oracle calls and offers more flexibility regarding the choice of the parameter set.

Some results on the cardinality of α -approximation sets are presented by Helfrich et al. [18]. First, even when given access to an exact oracle ALG_1 , no algorithm that has problem access only through this oracle can approximate the optimal cardinality B better than by a factor of $p + 1$. For $p = 1$, Bazgan et al. [17] describe an algorithm that can achieve the best-possible factor of 2. Second, for $p = 3$ and a routine ALG_β with $\beta > 1$, the cardinality B cannot be approximated within any constant factor $L \in \mathbb{N}$ by an algorithm that only has access to this routine.

As mentioned in Section 2.1, a survey of approximation algorithms for multi-objective optimization problems is given by Herzel et al. [35]. Every α -convex approximation algorithm mentioned therein can also be used for parametric optimization. We refrain from listing these algorithms here again. A recent algorithm for α -convex approximation is developed by Helfrich et al. [4]. It builds upon the dual Benson algorithm from multi-objective optimization. In contrast to the grid-based approaches, it chooses the parameter values where ALG_β is called in an adaptive fashion. A computational study in [4] shows that this adaptive strategy outperforms other general approximation algorithms on a set of generated instances for parametric knapsack and metric traveling salesman problems.

3.5 Parametric mixed integer linear programming

A *mixed integer linear program* (MILP) is a problem of the form

$$\begin{aligned} \min \quad & c^T x + d^T y \\ \text{s.t.} \quad & Ax + By \geq b \\ & x \in \mathbb{R}^{n_{\mathbb{R}}}, y \in \mathbb{Z}^{n_{\mathbb{Z}}} \end{aligned} \quad (1)$$

where $c \in \mathbb{R}^{n_{\mathbb{R}}}$, $d \in \mathbb{R}^{n_{\mathbb{Z}}}$, $A \in \mathbb{R}^{m_{\mathbb{R}} \times n_{\mathbb{R}}}$, $B \in \mathbb{R}^{m_{\mathbb{Z}} \times n_{\mathbb{Z}}}$, and $b \in \mathbb{R}^{m_{\mathbb{R}} + m_{\mathbb{Z}}}$ for some integers $m_{\mathbb{R}}, m_{\mathbb{Z}}, n_{\mathbb{R}}, n_{\mathbb{Z}} \geq 0$. If $n_{\mathbb{Z}} = 0$, the problem is a *linear program* (LP), and if $n_{\mathbb{R}} = 0$, it is an *integer linear program* (ILP). Parameter dependencies may appear in the objective function, the constraint matrix, or the right-hand side vector.

Note that all the specific problems discussed in Section 4 can also be formulated as (multi-)parametric MILPs. In this section, we take a short look at the case where a problem

is explicitly formulated as in (1). As this is an extensively researched area of parametric optimization, a complete coverage is out of scope for this survey. Instead of including every relevant reference, we point to some selected surveys, which together provide a comprehensive overview.

For parametric LPs, an introduction to several parametric simplex methods is provided by Gal [20]. Surveys covering more recent developments are presented by Pistikopoulos et al. [1] and Oberdieck et al. [2]. In particular, Tables 1 and 2 in [1] cover milestones and key developments in multi-parametric programming (including multi-parametric LPs), and Section 2.1 in [2] specifically covers key techniques for multi-parametric LPs. An important result on the cardinality of optimal solution sets for parametric LPs is given by Murty [83]. A single-parametric LP with a parameter-dependent right-hand side vector is constructed, where $n_{\mathbb{R}}$ is even, $m_{\mathbb{R}} = n_{\mathbb{R}}/2$, and the cardinality of an optimal solution set is $2^{n_{\mathbb{R}}/2}$. Due to LP duality, this result also holds for parametric LPs where only the objective function is parameter-dependent. Furthermore, since every parametric LP is also a parametric MILP, the same lower bound applies to parametric MILPs in general.

For parametric (M)ILPs, two surveys covering early developments are provided by Geoffrion and Nauss [19] and Greenberg [21], with the latter including an extensive annotated bibliography. Moreover, the two already mentioned surveys [1, 2] summarize the relevant literature and provide some further introduction into key techniques for multi-parametric ILPs and MILPs (and some further problem classes such as parametric quadratic programs). Most relevant for our survey is Section 2.3 in Oberdieck et al. [2], which specifically covers techniques for multi-parametric MILPs. Software that can be used to solve parametric MILPs is provided in the *multi-parametric toolbox (mpt)* [84] and in the *parametric optimization toolbox (POP)* [85].

4 Specific Problems

In the following sections, we take a look at several parametric combinatorial optimization problems that are well-researched in the literature. These problems can, in theory, also be formulated as parametric MILPs and can then be solved by strategies from Section 3. In this section, we look at literature that uses problem-specific combinatorial properties to obtain complexity results and algorithms.

We discuss the shortest path problem in Section 4.1, the minimum cut (or maximum flow) problem in Section 4.2, the minimum cost flow problem in Section 4.3, the matroid problem and minimum spanning tree problem in Section 4.4, the knapsack problem in Section 4.5, and the matching and assignment problems in Section 4.6. We list bounds on the cardinality of B and provide an overview of important algorithms for each problem. The specific problems together with the associated notational conventions are summarized in Table 2 in Appendix A.

4.1 Parametric shortest path problems

A parametric shortest path problem consists of a directed connected graph $G = (V, A)$ with $|V| = n$ vertices, $|A| = m$ arcs, a parameter set $\Lambda \subseteq \mathbb{R}^p$, and parametric arc weight functions $w_a(\lambda) = w_0(a) + \sum_{i=1}^p \lambda_i \cdot w_i(a)$ for $\lambda \in \Lambda$ and $a \in A$. There are three well-known variants of parametric shortest path problems.

The first variant is the parametric *single-pair shortest path problem* (SPSP), where a source vertex $s \in V$ and a target vertex $t \in V$ are given. A feasible solution is a directed simple path from s to t , and the goal is to compute, for every $\lambda \in \Lambda$, a feasible s - t -path P with minimum total length $F_\lambda(x) = \sum_{a \in P} w_a(\lambda)$.

The second variant is the parametric *single-source shortest path problem* (SSSP), where only a source vertex $s \in V$ is given. A feasible solution is a set of paths from s to all other vertices in V , and the goal is to find, for every $\lambda \in \Lambda$, a set of paths such that each of them is a shortest path to its target vertex.

The third variant is the parametric *all-pairs shortest path problem* (APSP), where a feasible solution is a set of paths between all pairs of vertices. The goal is to find, for every $\lambda \in \Lambda$, a set of paths such that all pairs of vertices are connected by a shortest path.

4.1.1 Intractability results

Since even the cardinality of optimal solution sets for the single-parametric SPSP can be exponential in the size of the graph, parametric shortest path problems are intractable in general. This result is shown in several papers.

The first proof is given by Carstensen [10]. She proves that, even in the single-parametric setting, there are instances for which the cardinality of an optimal solution set is in $n^{\Omega(\log(n))}$. Carstensen [86] also proves that this lower bound is tight. A matching upper bound of $n^{\log(n) + O(1)}$ on the cardinality of an optimal solution set is attributed to Gusfield [28].

Mulmuley and Shah [87] simplify the proof from Carstensen [10] and extend it by taking into account the bit-lengths of the arc weight functions. There is a revised variant of this proof⁴ from Gajjar and Radhakrishnan [88]. Their result transfers the lower bound of $n^{\Omega(\log(n))}$ also to planar graphs. Thereby, they refute an earlier conjecture by Nikolova [89]. The proof idea of Gajjar and Radhakrishnan [88] is to inductively compose dense bipartite graphs, so that the number of vertices increases by a constant factor and the number of breakpoints of the optimal cost function increases by a linear factor. Meanwhile, these non-planar bipartite graphs can be simulated by a planar gadget, where each arc is replaced by a path consisting of up to n^2 arcs and the original weight is distributed among these arcs.

4.1.2 Tractability results for special cases

While parametric shortest path problems are, in general, intractable, some special cases allow polynomial-time algorithms.

Karp and Orlin [90] consider single-parametric SSSPs in which the parameter coefficient $w_1(a)$ is either -1 or 0 for each arc $a \in A$. The parameter set Λ is chosen such that there is no negative-cost cycle for any $\lambda \in \Lambda$. Two algorithms are provided in [90], one with a running time in $\mathcal{O}(n^3)$, the other in $\mathcal{O}(nm \log(n))$. The latter algorithm works by maintaining a shortest path tree that changes iteratively while increasing the parameter. Young et al. [91] improve the running time bound of this algorithm by using Fibonacci heaps [92] to $\mathcal{O}(nm + n^2 \cdot \log(n))$.

An algorithm for the multi-parametric generalization of the problem considered by Karp and Orlin [90] is provided by Séré et al. [93]. The running time of this algorithm, however, is polynomial in the number of arcs, but exponential in the number of vertices.

⁴ Gajjar and Radhakrishnan [88] remark that the proof from Mulmuley and Shah [87] “is rather cryptic and has errors that throw the reader off”.

The parametric shortest path algorithms from Karp and Orlin [90] and Young et al. [91] are used by Erickson [94] to solve the non-parametric maximum flow problem in planar graphs. The maximum flow problem is interpreted as a single-parametric shortest path problem in the dual graph with parameter coefficients $w_1(a) \in \{-1, 0, 1\}$ for all arcs $a \in A$. Erickson [94] provides an algorithm that solves this parametric problem in $\mathcal{O}(n \log(n))$. This running time bound does not hold if the algorithm is used for more general parametric shortest path problems [94].

Correa et al. [95] tackle the issue of fare evasion in transit networks by building on results for the parametric shortest path problem. For single-parametric shortest path problems, they show that there are at most $m - 1$ breakpoints on series-parallel graphs.

4.1.3 Results for related shortest path problems

Apart from Correa et al. [95], there are also other optimization papers that use the connection to parametric shortest path problems to achieve new results for related problems.

Nikolova et al. [96] give an exact $n^{\Theta(\log n)}$ algorithm for the normally distributed stochastic shortest path problem, based on quasi-convex maximization. They use the connection between computational geometry and parametric optimization.

Chakraborty et al. [97] construct two-phase algorithms for single-parametric SSSPs. In the first phase of such an algorithm, a special data structure is created. In the second phase, this structure is used as an oracle that takes any $\lambda \in \Lambda$ for which there is no negative cycle as input and returns an optimal solution of $\Pi(\lambda)$. For single-parametric SSSPs, an algorithm is presented where the first phase runs in $\tilde{\mathcal{O}}(n^4)$ and queries in the second phase can be answered in $\mathcal{O}(m + n \log n)$. Another algorithm for p -parametric SPSPs achieves a running time in $\mathcal{O}(n^{\log n})$ for the first phase and in $\mathcal{O}(\log^2 n)$ for every second phase query. The latter algorithm can be generalized for arc weights that are polynomial functions of λ .

Foschini et al. [98] analyze the complexity of time-dependent shortest path problems by utilizing connections to parametric shortest path problems.

4.1.4 Approximation methods

Concerning the approximation of parametric shortest path problems, we are only aware of literature from multi-objective optimization.

For the bi-objective shortest path problem, Diakonikolas [45] gives a lower bound on the cardinality of approximation sets that can be computed in polynomial time. Transferred to the parametric setting, the theorem states that there is no polynomial-time $(1 + \varepsilon)$ -approximation algorithm that can approximate the cardinality of the smallest $(1 + \varepsilon)$ -approximation set to a factor smaller than 2, unless $P = NP$. An algorithm that achieves the best-possible factor of 2 can be obtained from the general approximation algorithm given by Bazgan et al. [17].

For an overview of approximation algorithms for multi-objective shortest path problems, we refer to the recent paper by Bökler and Chimani [99] and the general survey by Herzel et al. [35].

4.2 Parametric minimum cut (maximum flow) problems

A *parametric minimum cut problem* (sometimes also *parametric minimum s - t -cut problem*) consists of a directed graph $G = (V, A)$ with $|V| = n$ vertices and $|A| = m$ arcs, a source vertex $s \in V$, a target vertex $t \in V$ with $s \neq t$, a parameter set $\Lambda \subseteq \mathbb{R}^p$, and parametric

arc capacity functions $c_a(\lambda) = c_0(a) + \sum_{i=1}^p \lambda_i \cdot c_i(a)$ for $\lambda \in \Lambda$ and $a \in A$. A feasible solution $x \subseteq \{0, 1\}^m$ represents a set of arcs that partitions V into two disjoint subsets S with $s \in S$ and $V \setminus S$ with $t \in V \setminus S$, a so-called s - t -cut. The goal is to find, for every $\lambda \in \Lambda$, an s - t -cut $x \subseteq A$ with minimum total arc capacity $C_\lambda(x) = \sum_{a \in x} c_a(\lambda)$.

Due to the well-known max-flow-min-cut theorem (see [100]), intractability results for the parametric minimum cut problem can be applied to the parametric maximum flow problem, and vice versa.

4.2.1 Intractability results and approximation methods

The first proof of intractability for the single-parametric minimum cut problem is given by Carstensen [10] (see Section 4.1.1). However, the instance constructed in [10] includes negative arc capacities. When negative arc capacities are allowed, the minimum cut problem is NP-hard. Therefore, the proof of intractability by Mulmuley [101] provides a stronger result, as it applies to parametric minimum cut problems with non-negative arc capacities in at least one parameter interval. Any problem with a fixed parameter value in this interval can be solved in polynomial time, but there can still be superpolynomially many breakpoints within the interval. Gajjar and Radhakrishnan [88] tighten this result further: They construct the planar dual of the graph used for proving intractability of the shortest path problem. This instance can be used to verify that, for every $k \in \mathbb{N}$, there is an instance of the single-parametric minimum cut problem consisting of non-negative integer arc capacities and a planar graph with a number of vertices polynomial in k , bounded bit lengths of the arc capacity coefficients, and a linear capacity assignment to the arcs such that there are $k^{\Omega(\log(k))}$ breakpoints in the optimal cost curve.

Using a graph from Zadeh [102], Ruhe [12] proves intractability also for the generalized parametric maximum flow problem where the flow along each arc $(i, j) \in A$ is scaled by a multiplier $g_{ij} > 0$. Here, the goal is to find, for every $\lambda \in \Lambda$, a generalized s - t -flow such that the amount of flow reaching the target vertex t is maximized.

Allman et al. [103] show that 2-parametric minimum cut problems satisfying the so-called *specialized source-sink monotonicity* (sSSM) can still have exponentially many breakpoints. A graph with parametric arc capacities satisfies sSSM if, for each arc (s, j) , the arc capacity $c_{(s,j)}(\lambda_1, \lambda_2)$ is increasing in λ_1 and λ_2 , and, for all other arcs, the arc capacity is independent of λ_1 and λ_2 .

4.2.2 Exact solution methods for special cases

The parametric minimum cut problem is one of the most-researched problems in the parametric optimization literature. A wide range of applications is studied (e.g., [104–106]). Reviewing all the applications goes beyond the scope of this survey.

Instead, we focus on important exact solution methods for special cases of the problem. These special cases are based on the following properties:

Definition 6 (Nesting Property) An instance $(c, (G, s, t), \Lambda)$ has the *nesting property* if, for every sequence $\lambda_1 < \lambda_2 < \dots < \lambda_h$ of parameter values from Λ and $h \in \mathbb{N}$, there exist corresponding (*nested*) minimum cuts $(S_1, V \setminus S_1), (S_2, V \setminus S_2), \dots, (S_h, V \setminus S_h)$ such that $S_1 \subseteq S_2 \subseteq \dots \subseteq S_h$.

Definition 7 (Generalized Nesting Property) An instance $(c, (G, s, t), \Lambda)$ has the *generalized nesting property* if, for every sequence $\lambda_1 < \lambda_2 < \dots < \lambda_h$ of parameter values from Λ and $h \in \mathbb{N}$, there exist corresponding minimum cuts $(S_1, V \setminus S_1), (S_2, V \setminus S_2), \dots, (S_h, V \setminus S_h)$ such that $S_{\sigma(1)} \subseteq S_{\sigma(2)} \subseteq \dots \subseteq S_{\sigma(h)}$ for some permutation σ on $[h]$.

Definition 8 (Source-Sink Monotonicity (SSM)) An instance of the single-parametric minimum cut problem is *source-sink monotone* if, for every arc (i, j) , the arc capacity $c_{(i,j)}(\lambda)$ satisfies the condition

$$c_{(i,j)}(\lambda) \text{ is } \begin{cases} \text{non-decreasing in } \lambda & \text{if } i = s, j \neq t, \\ \text{non-increasing in } \lambda & \text{if } j = t, i \neq s, \\ \text{independent of } \lambda & \text{otherwise.} \end{cases}$$

Eisner and Severance [36] and Stone [107] show that minimum cuts of the single-parametric minimum cut problem satisfying SSM have the nesting property. The (generalized) nesting property implies that the optimal cost curve has at most $n - 1$ breakpoints [36]. For instances satisfying SSM, Gallo et al. [108] formulate a procedure that extends the push-relabel algorithm of Goldberg and Tarjan [109] and determines these breakpoints in the same asymptotic running time as a single maximum flow computation.

Arai et al. [110] consider parametric minimum cut problems where the capacities of the arcs incident to one specific vertex $u^* \neq s, t$ are non-decreasing functions of a parameter λ , whereas all other arc capacities are constant. They prove that, for such instances, the minimum cuts satisfy the generalized nesting property, which implies that the optimal cost curve has at most $n - 1$ breakpoints.

McCormick [111] extends the results of Gallo et al. [108] to more general multi-parametric minimum cut problems with piece-wise linear capacity functions and additional arcs not incident to s or t with parameter-dependent capacities. By again applying an extension of the push-relabel algorithm, the produced minimum cuts satisfy the nesting property, so the optimal cost curve has at most $n - 1$ breakpoints.

There is extensive literature that builds on these three fundamental papers and generalizes their theory. Extensions of Gallo et al. [108] include, e.g., the $\mathcal{O}(mn \log(n^2/m))$ maximum distance variant of push-relabel due to Gusfield and Tardos [112], Hochbaum's [113] $\mathcal{O}(mn \log(n))$ pseudoflow algorithm, and the divide-and-conquer algorithm from Tarjan et al. [114].

Brumelle et al. [115] comprises both the cases investigated by Gallo et al. [108] and by Arai et al. [110], and partially the case studied by McCormick [111]. They present an algebraic sufficient condition for the existence of a selection of optimal solutions in a parametric optimization problem that are totally ordered. However, the order of the critical regions that belong to the optimal solutions in this selection does not necessarily adhere to the same ordering.

Scutellà [116] extends the results of Gallo et al. [108] and Arai et al. [110] to networks in which the parametrization of the arc capacities can involve both the source and the target and one additional vertex u^* . She shows that the minimum cuts for such instances satisfy a relaxed form of the generalized nesting property, which again implies that this special case of a single-parametric minimum cut problem has at most $n - 1$ breakpoints. Finding these breakpoints needs asymptotically as much time as a single maximum flow computation and is based on an improvement of the maximum flow algorithm by Goldberg and Tarjan [109].

Granot et al. [117] identify three larger classes of parametric minimum cut problems satisfying the nesting property. Again, all minimum cuts can be computed in the same asymptotic time as a single minimum cut by using a specialized flow update strategy to move from one value of the parameter to the next.

If we turn the parametric problem into the related parametric search problem, there is further literature (see [117]). For example, the *discrete Newton algorithm* [118–122] efficiently solves the parametric search problem and can be extended to other parametric problems.

The literature covered so far only covers single-parametric minimum cut problems. Aissi et al. [123] show that, in the p -parametric *global* minimum cut problem, the number of cuts that become minimum at some point in the parameter space is in $\mathcal{O}(m^p n^2 \log^{p-1}(n))$. For the special case of two parameters, they give a tighter bound of $\mathcal{O}(n^3 \log(n))$ cuts that can be enumerated in $\tilde{\mathcal{O}}(m^2 n^4)$ time.

Karger [124] requires stronger conditions than Aissi et al. [123] to apply his results. The conditions are that every $\lambda \in \Lambda$ and $c_0(a), \dots, c_p(a)$ for all $a \in A$ are nonnegative. Under these conditions, the results are stronger and can be generalized in different directions. For instance, the $\mathcal{O}(n^{p+1})$ p -parametric *global* minimum cuts can be enumerated in $\tilde{\mathcal{O}}(mn^{p+1})$ time. He also gives a first generalization to single-parametric *global* minimum cuts with polynomial cost functions.

4.3 Parametric minimum cost flow problems

In the parametric *minimum cost flow problem* (MCF), a graph $G = (V, A)$ with $|V| = n$ vertices and $|A| = m$ arcs together with a source vertex $s \in V$, a target vertex $t \in V$ with $s \neq t$, a demand D , and a parameter set $\Lambda \subseteq \mathbb{R}^p$ are given. Additionally, for every arc $a \in A$, a lower capacity l_a , an upper capacity u_a , and a parametric arc cost function $c_a(\lambda) = c_0(a) + \sum_{i=1}^p \lambda \cdot c_i(a)$ are given. Every feasible solution $x: A \rightarrow \mathbb{R}$ represents an s - t -flow with flow value D that satisfies the capacity constraints $l_a \leq x(a) \leq u_a$ for all $a \in A$. The goal is to compute, for every $\lambda \in \Lambda$, a feasible s - t -flow x_λ that minimizes $\sum_{a \in A} x_\lambda(a) \cdot c_a(\lambda)$.

Based on a graph from Zadeh [102], Ruhe [12] shows that even single-parametric MCF instances may have optimal solution sets of exponential size.

4.3.1 Exact methods

For the MCF, every solution that is efficient in the multi-objective sense is also optimal for at least one parameter value in the parametric problem. Hence, many important results from the multi-objective literature are directly applicable to the parametric setting. Hamacher et al. [125] provide a comprehensive survey of bi-objective MCF algorithms. Many of these algorithms can be used for the single-parametric MCF as well. We refer to some important algorithms described therein. For detailed information on these, we refer directly to Hamacher et al. [125].

Network simplex algorithms can be applied to the parametric MCF. The idea is to pivot between basic feasible flows. Such an algorithm is described, for example, by Pulat et al. [126]. Lee and Simin Pulat [127] develop an algorithm that combines an out-of-kilter algorithm and a parametric linear programming procedure. Using these two approaches, Sedeño-Noda and González-Martín [128] provide an algorithm that outperforms them both in a computational study. Another algorithm by Sedeño-Noda and González-Martín [129] specializes the Eisner-Severance method for the parametric MCF, but is outperformed by their earlier algorithm from [128]. A primal-dual network simplex method is developed by Eusébio et al. [130]. In a

computational study, it outperforms a parametric simplex algorithm but is not compared to any of the aforementioned algorithms. The network simplex algorithm from Raith and Sedeño-Noda [131] uses a labeling strategy to avoid computing redundant minimum ratios. This leads to an output-polynomial running time in $\mathcal{O}(T_{\text{MCF}} + Bn(m + n \log n))$. In a computational study, the algorithm outperforms the Eisner-Severance method and a parametric simplex algorithm by Raith and Ehrgott [132].

In the parametric literature, recent interest is mostly focused on the single-parametric MCF with parametric lower bound constraints. Here, for every arc, the cost $c_a(\lambda)$ is a constant function of λ , but the lower capacities are of the form $l_a(\lambda) = l_0(a) - \lambda \cdot l_1(a)$.

Taking inspiration from generative linguistics and the behavior of neural networks, Sangeorzan et al. [133] present an algorithm for the MCF with single-parametric lower bounds. The algorithm recursively partitions the flows via a *preflow-pull* strategy. Its running time is in $\mathcal{O}(n^2 m^{1/2} + Bn)$ and can be sped up by using parallelization strategies.

In a series of papers, Ciurea and Parpalea [134], [135–138] develop several algorithms for the MCF with single-parametric lower bounds. An overview of the most important algorithmic results is provided in [138]. The first algorithm, given by Ciurea and Parpalea [134], covers only the special case of monotone parametric bipartite networks and is not always finite. An algorithm for the general case is developed in [135]. It iteratively computes the breakpoints and a minimum cost flow for each critical region in ascending order of the parameter values. Each iteration consists of modifying the network and then solving a maximum cost flow problem on it. Based on the resulting flow, the next breakpoint and minimum cost flow can be computed. It runs in time $\mathcal{O}(B T_{\text{MaxFlow}})$, where T_{MaxFlow} denotes the time to solve a single maximum flow problem.

In [136], Ciurea and Parpalea describe an algorithm that uses so-called *shortest conditional decreasing paths* on a residual network. The basic idea is to partition the parameter set and to repeatedly investigate each critical region and update it. Each investigation consist of finding shortest conditional decreasing paths to determine if an update is needed. The running time of this algorithm is in $\mathcal{O}(Bmn(B + mn))$.

A similar partitioning approach is taken in [137]. However, in the residual networks constructed there, it is sufficient to find any directed path since such a path must then necessarily be a conditional decreasing path as well. This leads to a better running time in $\mathcal{O}(Bn^2m)$.

4.4 Parametric matroid and spanning tree problems

Before we turn to the parametric matroid problem and some of its special cases, we define matroids as follows:

Definition 9 (Independent Sets and Matroids) A *matroid* consists of a finite ground set S and a family \mathcal{I} of subsets of S , called *independent sets*, with the following properties:

- I1) $\emptyset \in \mathcal{I}$
- I2) $\forall B \subseteq A \subseteq S : A \in \mathcal{I} \implies B \in \mathcal{I}$
- I3) $\forall A, B \in \mathcal{I} : |A| > |B| \implies \exists x \in A \setminus B : B \cup \{x\} \in \mathcal{I}$

An inclusion-wise maximal independent set of a matroid (S, \mathcal{I}) is called a *basis*, and the *rank* of a matroid is the cardinality of a basis.

Table 1 Upper and lower cardinality bounds for single-parametric matroid and minimum spanning tree problems. The function α denotes the inverse Ackermann function.

Problem ($p = 1$)	Setting & Notation	Cardinality Bound	Reference
General Matroid	n elements, rank k	$\Omega(nk^{1/3})$	[33]
		$\mathcal{O}(n \cdot \min(k, n - k)^{1/2})$	[139]
Graphic Matroid	$ V = n, E = m$	$\Omega(m\alpha(n))$	[33]
		$\Omega(m \log(n))$	[140]
		$\mathcal{O}(m\sqrt{n})$	[139]
		$\mathcal{O}(mn^{1/3})$	[141]
Uniform Matroid	n elements, rank k	$\Omega(n \log(k))$	[142]
		$\mathcal{O}(nk^{1/2})$	[143]
		$\mathcal{O}(nk^{1/2}/\log^*(k))$	[144]
		$\mathcal{O}(nk^{1/3})$	[141]

In the *parametric matroid problem*, a matroid (S, \mathcal{I}) with $|S| = n$, a parameter set $\Lambda \subseteq \mathbb{R}^p$, and parametric costs for the elements of S are given. The feasible solutions correspond to the bases of the matroid, and the objective function value $F_\lambda(x)$ of a basis x is given as the total cost of the elements of x . The goal is to obtain, for every $\lambda \in \Lambda$, a matroid basis x_λ of minimum total cost.

A special case of the parametric matroid problem is the parametric *minimum spanning tree problem* (MST) (also called the *parametric graphic matroid problem*). Here, a connected graph $G = (V, E)$ with $|V| = n$ vertices, $|E| = m$ edges, a parameter set $\Lambda \subseteq \mathbb{R}^p$, and parametric edge costs are given. The ground set is the set E of edges of G , and matroid bases correspond to spanning trees of G . Thus, the goal is to find, for every $\lambda \in \Lambda$, a spanning tree of minimum total edge cost.

Another special case is the *parametric uniform matroid problem*, where a ground set U_r of cardinality n is given, and matroid bases correspond to subsets containing exactly r elements. The goal is to find, for every $\lambda \in \Lambda$, a matroid basis x_λ of minimum total cost.

4.4.1 Cardinality bounds

Parametric matroid and minimum spanning tree problems are well-researched topics in the parametric literature. In contrast to many other problems, they always have an optimal solution set of polynomial cardinality in the single-parametric case. The most important results for matroids are summarized in Table 1.

For the parametric MST, the results of Seipp [145] for the multi-objective MST can be transferred to the parametric setting. They imply that, for every fixed number $p \geq 1$ of parameters, an optimal solution set of the parametric MST has a cardinality in $\mathcal{O}(m^{2(p+1)})$.

If property I3) is removed from Definition 9, the resulting problem is called *parametric independence systems problem*. Because intractable problems such as the parametric maximum matching problem and the parametric assignment problem are special cases of it, optimal solution sets can have a cardinality exponential in n [18].

4.4.2 Exact methods for the spanning tree problem

Fernández-Baca et al. [13] give an exact algorithm that solves the single-parametric MST in $\mathcal{O}(\min\{nm \log(n), T_{\text{MST}}(2n, n) \cdot B\})$ time, where $T_{\text{MST}}(m, n)$ denotes the time for computing a minimum spanning tree in a graph with m edges and n vertices. Their main idea is sparsification in order to pass through a reduced number of potential breakpoints. This gives a faster asymptotic running time than the Eisner-Severance method [13].

One motivation for improving the running times of algorithms for the parametric MST is the wide range of applications. Note that the construction procedure of Fernández-Baca et al. [13] can, among others [146], also be used for stochastic spanning trees [147]. Here, edge costs are not constant but random variables and the objective is to find an optimal spanning tree satisfying a certain probability constraint. After transforming the stochastic problem into the deterministic equivalent problem and including a positive parameter in the objective function, their parametric algorithm finds an optimal solution in $\mathcal{O}(n^6)$ time.

Agarwal et al. [148] develop deterministic and randomized algorithms for the single-parametric MST and the kinetic MST on general graphs as well as improved algorithms for special families such as minor-closed and planar graphs. In the (*structurally*) *kinetic minimum spanning tree problem*, edges can be inserted or removed by parameter changes. A multitude of well-known techniques are used to develop the algorithms therein, for example graph sparsification, convex hulls, and parametric search. For general graphs, algorithms with running time in $\mathcal{O}(n^{2/3} \log^{4/3}(n))$ (or randomized in $\mathcal{O}(n^{2/3} \log(n))$) are presented. If the graph is planar or from other minor-closed families of graphs, the running time bound for the parametric MST reduces to $\mathcal{O}(n^{1/4} \log^{3/2}(n))$ (or randomized to $\mathcal{O}(n^{1/4} \log(n))$), and for the kinetic MST to $\mathcal{O}(n^{1/2} \log^{3/2}(n))$ (or randomized to $\mathcal{O}(n^{1/2} \log(n))$).

4.4.3 Algorithms for different settings

In the multi-objective literature, an algorithm for the parametric MST is described by Seipp [145]. His approach works by computing the minimization diagram (see Section 3.3.1) in two steps: First, it uses some special properties of the parametric MST to find an arrangement of hyperplanes that subdivides Λ into polyhedra. Then, it iteratively checks for adjacent polyhedra whether they are part of the same critical region and merges such polyhedra until only the critical regions of a minimization diagram remain. No exact running time analysis is given, but a polynomial running time bound is shown.

There is also a lot of literature on parametric search or finding the shortest bottleneck edge for the parametric MST. However, this is not the focus of this survey and related literature can be found, for example, in [149–152].

Furthermore, there are other parametric problems related to the main problems discussed in this section. For example, Hausbrandt et al. [153] develops upper and lower bounds as well as algorithms for the parametric matroid interdiction problem. Due to the limited literature available, these problems are also not in our focus.

4.5 Parametric knapsack problems

The *parametric knapsack problem* (sometimes also called the *parametric 0-1-knapsack problem*) is defined by a item weight vector $w = (w_1, \dots, w_n)^T \in \mathbb{N}^n$, a budget $K \in \mathbb{N}$, a parameter set $\Lambda \subseteq \mathbb{R}^p$, and a parametric item profit vector $c(\lambda) = (c_1(\lambda), \dots, c_n(\lambda))^T \in \mathbb{R}^n$ with $c_i(\lambda) = c_0^i + \sum_{j=1}^p \lambda_j \cdot c_j^i$ for $i \in [n]$ and $\lambda \in \Lambda$. A feasible solution $x \in \{0, 1\}^n$ (a *packing*) represents a subset of selected items such that $w^T x \leq K$. The goal is to find, for every $\lambda \in \Lambda$, a feasible packing $x \in \{0, 1\}^n$ maximizing the total profit $c(\lambda)^T x$ of the selected items.

4.5.1 Intractability results and exact methods

Carstensen [10] is the first to address the parametric knapsack problem and shows that the cardinality of an optimal solution set for a single-parametric knapsack problem can be exponential in the number of items. This result even holds under the additional assumption that Λ is a compact interval in $\mathbb{R}_{>0}$ and that $c_0^i, c_1^i > 0$ for all $i \in [n]$. Carstensen [10] also shows, however, that the cardinality of an optimal solution set of any single-parametric *binary* integer program is linear in the number of variables when the cost function coefficients c_0^i and c_1^i are integers in $[-M, M]$ for some constant $M \in \mathbb{N}$ and all $i \in [n]$. In particular, this implies that the cardinality of an optimal solution set of the single-parametric knapsack problem is linear in the number of items under this condition.

An algorithm for the single-parametric knapsack problem is developed by Eben-Chaïme [154]. This algorithm combines the Eisner-Severance method with a classic dynamic programming approach for knapsack problems. Its running time is in $\mathcal{O}(KnB)$.

A problem closely related to the parametric knapsack problem is the *inverse-parametric knapsack problem*, where the goal is to find the smallest parameter for which the optimal objective value of a parametric knapsack problem is equal to a specific input value. For this problem, Burkard and Pferschy [155] provide a pseudo-polynomial algorithm based on Megiddo's method with a running time in $\mathcal{O}(n^2 K \log(K) / \log(n \log(K)))$.

4.5.2 Parametric approximation methods

Several approximation algorithms exist for the single-parametric knapsack problem. Giudici et al. [15] provide a PTAS for the single-parametric knapsack problem. The authors also describe how approximation algorithms of bi-objective optimization problems can be used for approximating single-parametric optimization problems. They use this result and the algorithm from Erlebach et al. [156] to derive an FPTAS for the case that $\Lambda = \mathbb{R}_{\geq 0}$ and all profit coefficients in the objective function are nonnegative.

Holzhauser and Krumke [14] describe an FPTAS that works on arbitrary parameter intervals and profit coefficients in the objective function. Furthermore, in contrast to the FPTAS from Giudici et al. [15], the running time is strongly polynomial. For any $\varepsilon > 0$, the algorithm finds a $(1 - \varepsilon)$ -approximation set with cardinality in $\mathcal{O}(n^{2/\varepsilon})$ and has a running time in $\mathcal{O}(n^{2/\varepsilon} \cdot T_{\Pi}(\varepsilon))$, where $T_{\Pi}(\varepsilon)$ denotes the running time of an FPTAS for the conventional knapsack problem with an approximation factor $1 + \varepsilon$.

Halman et al. [16] investigate a variant of the single-parametric knapsack problem in which the parameter dependence is in the weight vector instead of the objective function. They provide two FPTASs, one of which has a running time in $\mathcal{O}(n^{3/\varepsilon^2} \cdot \min\{\log^2(P), n^2\} \cdot \min\{\log(M), n \log(n/\varepsilon)/\log(n \log(n/\varepsilon))\})$, where $\varepsilon > 0$ determines the approximation precision, P is an upper bound on the optimal profit, and M is an upper bound on the reachable weight.

4.6 Parametric matching and assignment problems

In the *parametric assignment problem*, a bipartite, undirected graph $G = (U \cup V, E)$ with $|U| = |V| = n$ vertices and $|E| = m$ edges, a parameter set $\Lambda \subseteq \mathbb{R}^P$, and parametric edge weights $w_e(\lambda) = w_0(e) + \sum_{i=1}^P \lambda_i \cdot w_i(e)$ for $\lambda \in \Lambda$ and $e \in E$ are given. An assignment is a set of edges where every edge is between U and V , and no vertex from U or V is an endpoint more than once. The goal of the parametric assignment problem is to compute, for every $\lambda \in \Lambda$, a minimum weight assignment.

The parametric assignment problem is intractable, even in the single-parametric case. To prove this, Bazgan et al. [17] sketch how to use the non-parametric reduction of the shortest path problem to the assignment problem from Lawler [157]. The first proof of intractability is given by Carstensen [86] (referring to the reduction in [158]).

However, for the special case where the parameter coefficients $w_1(e)$ are either 0 or 1 for all $e \in E$ the single-parametric assignment problem becomes tractable. A polynomial-time algorithm is given by Gassner and Klinz [11]. It repeatedly uses the parametric shortest path algorithm of Young et al. [91] and its running time is in $\mathcal{O}(mn + n^2 \log(n))$. Rosenmann [159] uses the parametric algorithm from Gassner and Klinz [11] for computing the sequence of all k -cardinality assignments.

The parametric assignment problem can be seen as a special case of the parametric maximum matching problem. Hence, the (single-)parametric maximum matching problem is also intractable [18]. In the literature, the parametric maximum matching problem is rarely investigated. Weber [160] conducts a sensitivity analysis for the case where one edge weight depends on a single parameter. Gusfield and Irving [161] consider the single-parametric stable matching problem and develop an efficient algorithm by reducing it to the parametric minimum cut problem and using the method of Gallo et al. [108].

5 Conclusion

The literature on parametric optimization dates back over 60 years, and some contributions appear even earlier under different names such as sensitivity analysis [7]. Figure 5 visualizes the distribution of the literature in this paper over the years. Although it does not include papers that are solely concerned with practical applications, have somewhat different problem settings, or are simply surveyed in other literature reviews, Figure 5 shows the ongoing interest in parametric optimization.

In nearly all papers we surveyed, the parameter dependencies are linear and appear in the objective function. Parameter-dependent constraints are mainly discussed for the generic MILP case. But for specific problems, there are often structural properties that can be used to find cardinality bounds or develop specialized algorithms. This can be seen in some of the algorithms we surveyed, for example, for minimum cost flow and maximum flow problems or for the knapsack problem. For most other problems, however, such algorithms are rare.

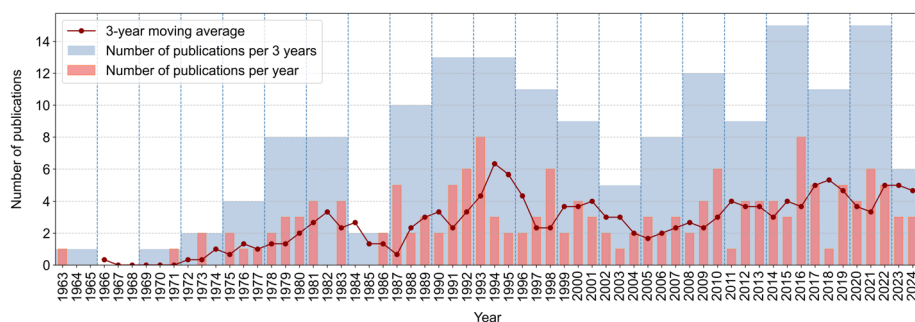


Fig. 5 Number of publications covered in this survey over time (per year, per 3 years, and 3-year moving average) from 1963 to 2024. The moving average for year t considers the years $t - 1$, $t - 2$, and $t - 3$.

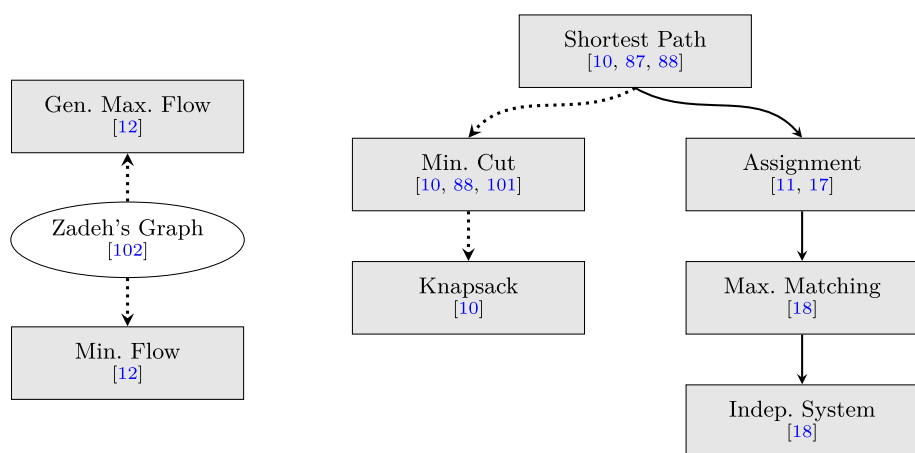


Fig. 6 Relations between intractability results from Section 4. A solid arc indicates a proof via reduction, a dotted arc indicates that constructions leading to optimal solution sets with superpolynomial cardinality are transferred into a different problem setting

The main challenge in parametric optimization is intractability and finding strategies to cope with it. Most parametric problems are intractable because of the cardinality of optimal solution sets. This can be the case even if the underlying non-parametric optimization problem is solvable in polynomial time, which is the case, for example, for parametric shortest path problems. Table 3 in Appendix B summarizes known bounds on the cardinality of optimal solution sets for all the specific problems considered in this survey.

For proving intractability, many papers use similar strategies. This is visualized in Figure 6: Either intractable parametric problems are reduced to the problem for which intractability is to be shown, or constructions that lead to optimal solution sets with superpolynomial cardinality are transferred into a different problem setting.

Still, some problems admit an optimal solution set of polynomial size. This has been shown mainly for matroid-based problems. Restricting the parameter coefficients to specific integers also often makes a problem tractable. Problems where this can be observed are shortest path problems, the assignment problem, and binary integer programs with bounded cost function coefficients.

When dealing with intractable problems, a way to still achieve polynomial running time is to find an approximation set instead of an optimal solution set. However, approximation algorithms are rarely found in the literature. We discussed some general approximation algorithms in Section 3.4. To the best of our knowledge, the only problem-specific approximation algorithms can be found for the knapsack problem. This demonstrates a gap in the literature – particularly compared to multi-objective optimization, where problem-specific approximation algorithms are available for a large variety of problems [35].

With respect to methods for computing an optimal solution set, algorithms for specific problems are often adaptations of general strategies. For example, the algorithms of Sedeño-Noda and González-Martín [129] for the parametric minimum cost flow problem, of Fernández-Baca et al. [13] for the parametric spanning tree problem, and of Eben-Chaime [154] for the knapsack problem are tailored variants of the Eisner-Severance method, while Megiddo's method is used by Burkard and Pferschy [155] for the parametric knapsack problem and by Agarwal et al. [148] for the parametric spanning tree problem. These adaptation strategies further highlight the value of finding problem-agnostic methods. Here, research areas related to parametric optimization may offer new opportunities. For example, the immediate connection between parametric optimization and multi-objective optimization is used numerous times in this survey to transfer (in-)tractability results and algorithms, but similar connections might also exist to other research areas.

Author Contributions **Levin Nemesch:** Conceptualization, Methodology, Investigation, Literature Review, Data Curation, Writing - Original Draft, Writing - Review and Editing, Visualization. **Stefan Ruzika:** Conceptualization, Methodology, Writing - Review and Editing, Supervision, Project Administration, Funding acquisition. **Clemens Thielen:** Conceptualization, Methodology, Writing - Review and Editing, Supervision, Project Administration, Funding acquisition. **Alina Wittmann:** Conceptualization, Methodology, Investigation, Literature Review, Data Curation, Writing - Original Draft, Writing - Review and Editing, Visualization.

Funding Open Access funding enabled and organized by Projekt DEAL. This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project number 508981269.

Declarations

Competing Interests The authors have no relevant financial or non-financial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Appendix A – Overview of Problem Classes

Table 2 Overview of the problem classes considered in this survey. Each problem is given a parameter set $\Lambda \subseteq \mathbb{R}^p$. $G = (V, A)$ is a directed and $G = (V, E)$ an undirected graph with $|V| = n$ vertices and $|A| = m$ arcs or $|E| = m$ edges

Parametric Problem	Instance	Task
Single-Pair Shortest Path (SPSP)	$G = (V, A)$ connected, $s, t \in V$ with $s \neq t$, param. arc weight functions $w_a(\lambda) = w_0(a) + \sum_{i=1}^p \lambda_i w_i(a)$ for $\lambda \in \Lambda$ & $a \in A$	Find for every $\lambda \in \Lambda$ a simple s - t -path x with min. total length $F_\lambda(x) = \sum_{a \in x} w_a(\lambda)$.
Single-Source Shortest Path (SSSP)	$G = (V, A)$ connected, $s \in V$, param. arc weight functions $w_a(\lambda) = w_0(a) + \sum_{i=1}^p \lambda_i w_i(a)$ for $\lambda \in \Lambda$ & $a \in A$	Find for every $\lambda \in \Lambda$ a set x of paths from s to all $v \in V$, each having min. total length.
All-Pairs Shortest Path (APSP)	$G = (V, A)$ connected, param. arc weight functions $w_a(\lambda) = w_0(a) + \sum_{i=1}^p \lambda_i w_i(a)$ for $\lambda \in \Lambda$ & $a \in A$	Find for every $\lambda \in \Lambda$ a set x of paths between all $v, w \in V$, $v \neq w$, each having min. total length.
Min. Cut	$G = (V, A)$, $s, t \in V$ with $s \neq t$ and param. arc capacity functions $c_a(\lambda) = c_0(a) + \sum_{i=1}^p \lambda_i c_i(a)$ for $\lambda \in \Lambda$ & $a \in A$	Find for every $\lambda \in \Lambda$ an s - t -cut $x \subseteq \{0, 1\}^m$ with min. total arc capacity $C_\lambda(x) = \sum_{a \in x} c_a(\lambda)$.
Min. Cost Flow (MCF)	$G = (V, A)$, $s, t \in V$ with $s \neq t$, demand D , lower capacity l_a , upper capacity u_a , param. arc cost function $c_a(\lambda) = c_0(a) + \sum_{i=1}^p \lambda_i c_i(a)$	Find for every $\lambda \in \Lambda$ an s - t -flow x with flow value D satisfying $l_a \leq x(a) \leq u_a$ for all $a \in A$ and min. $\sum_{a \in A} x(a) c_a(\lambda)$
(Uniform) Matroid	Matroid (S, \mathcal{I}) (bases correspond to subsets containing exactly r elements), param. costs for $s \in S$	Find for every $\lambda \in \Lambda$ a matroid basis x of min. total cost $F_\lambda(x)$.
Max. of Independence Systems	Independence system (S, \mathcal{I}) , param. costs for $s \in S$	Find for every $\lambda \in \Lambda$ an independent set x of max. total cost $F_\lambda(x)$.
Min. Spanning Tree (MST)	$G = (V, E)$ connected, param. edge costs	Find for every $\lambda \in \Lambda$ a spanning tree x of min. total edge cost $F_\lambda(x)$.
Knapsack	Item weight vector $w \in \mathbb{N}^n$, budget $K \in \mathbb{N}$, param. item profit vector $(c_1(\lambda), \dots, c_n(\lambda))^T \in \mathbb{R}^n$ with $c_i(\lambda) = c_0^i + \sum_{j=1}^p \lambda_j c_j^i$ for $i \in [n]$ & $\lambda \in \Lambda$	Find for every $\lambda \in \Lambda$ a packing $x \in \{0, 1\}^n$ satisfying $w^T x \leq K$ and max. total profit $c(\lambda)^T x$.
Assignment	$G = (U \cup V, E)$ bipartite, param. edge weights $w_e(\lambda) = w_0(e) + \sum_{i=1}^p \lambda_i w_i(e)$ for $\lambda \in \Lambda$ & $e \in E$	Find for every $\lambda \in \Lambda$ an assignment x of min. total weight $W_\lambda(x) = \sum_{e \in x} w_e(\lambda)$.
(MI)LP	$c_0, c_1, \dots, c_p \in \mathbb{R}^{n_{\mathbb{R}}}$, $d_0, d_1, \dots, d_p \in \mathbb{R}^{n_{\mathbb{Z}}}$, $A \in \mathbb{R}^{m_{\mathbb{R}} \times n_{\mathbb{R}}}$, $B \in \mathbb{R}^{m_{\mathbb{Z}} \times n_{\mathbb{Z}}}$, $b \in \mathbb{R}^{m_{\mathbb{R}} + m_{\mathbb{Z}}}$ for some $m_{\mathbb{R}}, m_{\mathbb{Z}} \geq 0$, $n_{\mathbb{R}} \geq 0$ ($= 0$ for ILP), $n_{\mathbb{Z}} \geq 0$ ($= 0$ for LP)	min $(c_0 + \sum_{i=1}^p \lambda_i c_i)^T x$ + $(d_0 + \sum_{i=1}^p \lambda_i d_i)^T y$ s.t. $Ax + By \geq b$, $x \in \mathbb{R}^{n_{\mathbb{R}}}$, $y \in \mathbb{Z}^{n_{\mathbb{Z}}}$

Appendix B – Cardinality Bounds

Table 3 Comparison of best upper/lower cardinality bounds and algorithmic results for specific parametric optimization problems

Parametric Problem		Cardinality Bounds		Reference	Algorithmic Results	
Problem	p	Setting & Notation	Cardinality Bound		Running Time	
0-1-Knapsack	1	n items	$2^n - 1$ (lower bound)	[10]	$\mathcal{O}(KnB)$ for K knapsack budget, n items, B break-points [154]	
	1	n items, f_i linear with integer weights bounded by $-C_i$ and C_i	$\min\{(2C_1 + 1)n, (4C_0 + 2)n\}$ (upper bound)	[10]		
Assignment	1	$ V + U = 2n$ vertices	$2^{\Omega(\log^2(n))}$	[17]	$\mathcal{O}(mn + n^2 \log n)$ for special case $F_\lambda(x) = \sum_{e \in E} (a_e + \lambda b_e) \cdot x_e$ with $b_e \in \{-1, 0\} \forall e \in E$ [11]	
	1	$ V + U = 2n$ vertices	$2^{\mathcal{O}(n)}$	[162]		
Min. Cut	1	$n + 2$ vertices	$2^n - 1$ (lower bound)	[10]	$\tilde{\mathcal{O}}(mn^{1+p})$ for special case of finding g_{global} min. cuts in p dimensions with nonneg. weights and parameters [124]	
	1	planar G , polynomial number vertices in n	$2^{\Omega(\log^2(n))}$	[88]		
Shortest Path	2	$n + 2$ vertices, fulfilling SSM	$2^n - 1$ (lower bound)	[103]	$\mathcal{O}(1)$ max. flow computations for special case of single-parametric problem fulfilling SSM [108] $\mathcal{O}(mn + n^2 \log n)$ for single-source case, $F_\lambda(x) = \sum_{e \in E} (a_e + \lambda b_e) \cdot x_e, b_e \in \{-1, 0\} \forall e \in E$ [91]	
	1	$n + 2$ vertices, fulfilling SSM	$n - 1$ (upper bound)	[36, 107]		
	1	layered & planar G , n vertices	$2^{\Omega(\log^2(n))}$	[88]		
	1	n vertices	$n \log n + \mathcal{O}(1)$	[86]		

Table 3 continued

Parametric Problem		Cardinality Bounds		Algorithmic Results	
Problem	p	Setting & Notation	Cardinality Bound	Reference	Running Time
General Matroid	1	n elements, rank k	$\mathcal{O}(n \cdot \min(k, n - k)^{1/2})$	[139]	$\mathcal{O}(n^{1/4} \log^{3/2}(n) B)$ for planar graphs, B breakpoints [148]
	1	n elements, rank k	$\Omega(nk^{1/3})$	[33]	
	1	$ V = n, E = m$	$\mathcal{O}(m\sqrt{n})$	[139]	
	1	$ V = n, E = m$	$\mathcal{O}(mn^{1/3})$	[141]	
	1	$ V = n, E = m$, inverse Ackermann function α	$\Omega(m\alpha(n))$	[33]	
Uniform Matroid	1	$ V = n, E = m$	$\Omega(m \log n)$	[140]	$\mathcal{O}(n^2 m^{1/2} + Bn)$ for B breakpoints [133] $\mathcal{O}(nm \log(n^2/m) \log(nC))$ for C max. arc cost [163]
	$p - 1$	$ E = m$	$\mathcal{O}(m^{2p})$	[145]	
	1	n elements, rank k	$\mathcal{O}(nk^{1/2})$	[143]	
	1	n elements, rank k	$\mathcal{O}(nk^{1/2}/\log^*(k))$	[144]	
	1	n elements, rank k	$\Omega(n \log(k))$	[142]	
Min. Cost Flow	1	n elements, rank k	$\mathcal{O}(nk^{1/3})$	[141]	
	1	$n + 2$ vertices	$2^n - 1$ (lower bound)	[12]	
LP	1	$x \in \mathbb{R}^{n_1}, y \in \mathbb{Z}^{n_2}, n_1$ even, $m_1 = n_1/2$ for $m_1 + m_2$ dim. of RHS	$2^{n_1/2}$ (lower bound)	[83]	

References

1. Pistikopoulos, E.N., Dominguez, L., Panos, C., Kouramas, K., Chinchuluun, A.: Theoretical and algorithmic advances in multi-parametric programming and control. *Computational Management Science* **9**(2), 183–203 (2012). <https://doi.org/10.1007/s10287-012-0144-4>
2. Oberdieck, R., Diangelakis, N.A., Nascu, I., Papathanasiou, M.M., Sun, M., Avraamidou, S., Pistikopoulos, E.N.: On multi-parametric programming and its applications in process systems engineering. *Chemical Engineering Research and Design* **116**, 61–82 (2016). <https://doi.org/10.1016/j.cherd.2016.09.034>
3. Jenkins, L.: Using parametric integer programming to plan the mix of an air transport fleet. *Information Systems and Operational Research* **25**(2), 117–135 (1987). <https://doi.org/10.1080/03155986.1987.11732033>
4. Helfrich, S., Ruzika, S., Thielen, C.: Efficiently constructing convex approximation sets in multiobjective optimization problems. *INFORMS Journal on Computing* (2024). <https://doi.org/10.1287/ijoc.2023.0220>
5. Fontaine, D., Michel, L., Hentenryck, P.V.: Constraint-based lagrangian relaxation. In: *Proceedings of the 20th International Conference on Principles and Practice of Constraint Programming (CP)*. Lecture Notes in Computer Science, vol. 8656, pp. 324–339 (2014). https://doi.org/10.1007/978-3-319-10428-7_25
6. Avraamidou, S., Pistikopoulos, E.N.: Adjustable robust optimization through multi-parametric programming. *Optimization Letters* **14**(4), 873–887 (2020). <https://doi.org/10.1007/s11590-019-01438-5>
7. Gal, T., Greenberg, H.J. (eds.): *Advances in Sensitivity Analysis and Parametric Programming*, 1st edn. International Series in Operations Research & Management Science, vol. 6. Springer, New York (1997). <https://doi.org/10.1007/978-1-4615-6103-3>
8. Avraamidou, S., Pistikopoulos, E.N.: Multi-parametric global optimization approach for tri-level mixed-integer linear optimization problems. *Journal of Global Optimization* **74**(3), 443–465 (2019). <https://doi.org/10.1007/s10898-018-0668-4>
9. Tokuyama, T.: Combinatorics on arrangements and parametric matroids: A bridge between computational geometry and combinatorial optimization. *IEICE Transactions on Information and Systems* **E83-D**(3), 362–371 (2000)
10. Carstensen, P.J.: Complexity of some parametric integer and network programming problems. *Mathematical Programming* **26**, 64–75 (1983). <https://doi.org/10.1007/BF02591893>
11. Gassner, E., Klinz, B.: A fast parametric assignment algorithm with applications in max-algebra. *Networks* **55**(2), 61–77 (2010). <https://doi.org/10.1002/net.20288>
12. Ruhe, G.: Complexity results for multicriterial and parametric network flows using a pathological graph of Zadeh. *Zeitschrift für Operations Research* **32**, 9–27 (1988). <https://doi.org/10.1007/BF01920568>
13. Fernández-Baca, D., Slutzki, G., Eppstein, D.: Using sparsification for parametric minimum spanning tree problems. In: *Proceedings of the 5th Scandinavian Workshop on Algorithm Theory (SWAT)*. Lecture Notes in Computer Science, vol. 1097, pp. 149–160 (1996). https://doi.org/10.1007/3-540-61422-2_128
14. Holzhauser, M., Krumke, S.O.: An FPTAS for the parametric knapsack problem. *Information Processing Letters* **126**, 43–47 (2017). <https://doi.org/10.1016/j.ipl.2017.06.006>
15. Giudici, A., Halffmann, P., Ruzika, S., Thielen, C.: Approximation schemes for the parametric knapsack problem. *Information Processing Letters* **120**, 11–15 (2017). <https://doi.org/10.1016/j.ipl.2016.12.003>
16. Halman, N., Holzhauser, M., Krumke, S.O.: An FPTAS for the knapsack problem with parametric weights. *Operations Research Letters* **46**(5), 487–491 (2018). <https://doi.org/10.1016/j.orl.2018.07.005>
17. Bazgan, C., Herzel, A., Ruzika, S., Thielen, C., Vanderpooten, D.: An approximation algorithm for a general class of parametric optimization problems. *Journal of Combinatorial Optimization* **43**, 1328–1358 (2022). <https://doi.org/10.1007/s10878-020-00646-5>
18. Helfrich, S., Herzel, A., Ruzika, S., Thielen, C.: An approximation algorithm for a general class of multi-parametric optimization problems. *Journal of Combinatorial Optimization* **44**, 1459–1494 (2022). <https://doi.org/10.1007/s10878-022-00902-w>
19. Geoffrion, A.M., Nauss, R.: Parametric and postoptimality analysis in integer linear programming. *Management Science* **23**(5), 453–466 (1977). <https://doi.org/10.1287/mnsc.23.5.453>
20. Gal, T.: *Postoptimal Analyses, Parametric Programming, and Related Topics: Degeneracy, Multicriteria Decision Making, Redundancy*, 2nd edn. Walter de Gruyter, Berlin, Germany (1995). <https://doi.org/10.1515/9783110871203>
21. Greenberg, H.J.: An annotated bibliography for post-solution analysis in mixed integer programming and combinatorial optimization. In: Woodruff, D.L. (ed.) *Advances in Computational and Stochastic Optimization, Logic Programming, and Heuristic Search: Interfaces in Computer Science and Operations Research*, pp. 97–147. Springer, New York (1998). https://doi.org/10.1007/978-1-4757-2807-1_4

22. Marshall, F.L.: The lagrangian relaxation method for solving integer programming problems. *Management Science* **27**(1), 1–18 (1981). <https://doi.org/10.1287/mnsc.27.1.1>
23. Bryson, N.: Parametric programming and lagrangian relaxation: The case of the network problem with a single side-constraint. *Computers & Operations Research* **18**(2), 129–140 (1991). [https://doi.org/10.1016/0305-0548\(91\)90084-5](https://doi.org/10.1016/0305-0548(91)90084-5)
24. Bryson, N.: A parametric programming methodology to solve the lagrangian dual for network problems with multiple side-constraints. *Computers & Operations Research* **20**(5), 541–552 (1993). [https://doi.org/10.1016/0305-0548\(93\)90017-D](https://doi.org/10.1016/0305-0548(93)90017-D)
25. Gabrel, V., Murat, C., Thiele, A.: Recent advances in robust optimization: An overview. *European Journal of Operational Research* **235**(3), 471–483 (2014). <https://doi.org/10.1016/j.ejor.2013.09.036>
26. Chuong, T.D.: Exact relaxations for parametric robust linear optimization problems. *Operations Research Letters* **47**(2), 105–109 (2019). <https://doi.org/10.1016/j.orl.2019.01.005>
27. Antczak, T.: Parametric approach for approximate efficiency of robust multiobjective fractional programming problems. *Mathematical Methods in the Applied Sciences* **44**(14), 11211–11230 (2021). <https://doi.org/10.1002/mma.7482>
28. Gusfield, D.: Sensitivity analysis for combinatorial optimization. PhD thesis, Electronics Research Laboratory, University of California, Berkeley (1980)
29. Gal, T., Wolf, K.: Stability in vector maximization—A survey. *European Journal of Operational Research* **25**(2), 169–182 (1986). [https://doi.org/10.1016/0377-2217\(86\)90083-4](https://doi.org/10.1016/0377-2217(86)90083-4)
30. Dempe, S., Zemkoho, A.: Bilevel Optimization: Advances and Next Challenges. Springer Optimization and Its Applications, vol. 161. Springer, Cham, Switzerland (2020). <https://doi.org/10.1007/978-3-030-52119-6>
31. Goyal, V., Rani, N., Gupta, D.: Parametric approach to quadratically constrained multi-level multi-objective quadratic fractional programming. *OPSEARCH* **58**(3), 557–574 (2021). <https://doi.org/10.1007/s12597-020-00497-y>
32. Avraamidou, S., Pistikopoulos, E.N.: Multi-level Mixed-Integer Optimization: Parametric Programming Approach. De Gruyter, Berlin, Boston (2022). <https://doi.org/10.1515/9783110760316>
33. Eppstein, D.: Geometric lower bounds for parametric matroid optimization. *Discrete & Computational Geometry* **20**(4), 463–476 (1998). <https://doi.org/10.1007/PL00009396>
34. Ehrgott, M.: Multicriteria Optimization, 2nd edn. Springer, Berlin, Heidelberg (2005). <https://doi.org/10.1007/3-540-27659-9>
35. Herzel, A., Ruzika, S., Thielen, C.: Approximation methods for multiobjective optimization problems: A survey. *INFORMS Journal on Computing* **33**(4), 1284–1299 (2021). <https://doi.org/10.1287/ijoc.2020.1028>
36. Eisner, M.J., Severance, D.G.: Mathematical techniques for efficient record segmentation in large shared databases. *Journal of the ACM* **23**(4), 619–635 (1976). <https://doi.org/10.1145/321978.321982>
37. Nauss, R.: Parametric integer programming. PhD thesis, Western Management Science Institute, University of California, Los Angeles (1975)
38. Radke, M.: Sensitivity analysis in discrete optimization. PhD thesis, Western Management Science Institute, University of California, Los Angeles (1975)
39. Csirmaz, L.: Inner approximation algorithm for solving linear multiobjective optimization problems. *Optimization* **70**(7), 1487–1511 (2021). <https://doi.org/10.1080/02331934.2020.1737692>
40. Aneja, Y.P., Nair, K.P.K.: Bicriteria transportation problem. *Management Science* **25**(1), 73–78 (1979). <https://doi.org/10.1287/mnsc.25.1.73>
41. Cohon, J.L.: Multiobjective Programming and Planning. Mathematics in science and engineering, vol. 140. New York Academic Press, New York (1978)
42. Dial, R.B.: A model and algorithm for multicriteria route-mode choice. *Transportation Research Part B: Methodological* **13**(4), 311–316 (1979). [https://doi.org/10.1016/0191-2615\(79\)90024-9](https://doi.org/10.1016/0191-2615(79)90024-9)
43. Burkard, R.E., Hamacher, H.W., Rote, G.: Sandwich approximation of univariate convex functions with an application to separable convex programming. *Naval Research Logistics* **38**(6), 911–924 (1991). <https://doi.org/10.1002/nav.3800380609>
44. Rote, G.: The convergence rate of the sandwich algorithm for approximating convex functions. *Computing* **48**(3), 337–361 (1992). <https://doi.org/10.1007/BF02238642>
45. Diakonikolas, I.: Approximation of multiobjective optimization problems. PhD thesis, Graduate School of Arts and Sciences, Columbia University (2011). <https://doi.org/10.7916/D80K2GR9>
46. Daskalakis, C., Diakonikolas, I., Yannakakis, M.: How good is the chord algorithm? *SIAM Journal on Computing* **45**(3), 811–858 (2016). <https://doi.org/10.1137/13093875X>
47. Gusfield, D.: Parametric combinatorial computing and a problem of program module distribution. *Journal of the ACM* **30**(3), 551–563 (1983). <https://doi.org/10.1145/2402.322391>

48. Megiddo, N.: Combinatorial optimization with rational objective functions. *Mathematics of Operations Research* **4**(4), 414–424 (1979). <https://doi.org/10.1287/moor.4.4.414>
49. Megiddo, N.: Applying parallel computation algorithms in the design of serial algorithms. *Journal of the ACM* **30**(4), 852–865 (1983). <https://doi.org/10.1145/2157.322410>
50. Cole, R.: Slowing down sorting networks to obtain faster sorting algorithms. *Journal of the ACM* **34**(1), 200–208 (1987). <https://doi.org/10.1145/7531.7537>
51. Oostrum, R., Veltkamp, R.C.: Parametric search made practical. *Computational Geometry* **28**(2), 75–88 (2004). <https://doi.org/10.1016/j.comgeo.2004.03.006>
52. Cohen, E., Megiddo, N.: Maximizing concave functions in fixed dimension. In: Pardalos, P.M. (ed.) *Complexity in Numerical Optimization*, pp. 74–87. World Scientific, Singapore (1993). https://doi.org/10.1142/9789814354363_0005
53. Toledo, S.: Maximizing non-linear concave functions in fixed dimension. In: Pardalos, P.M. (ed.) *Complexity in Numerical Optimization*, pp. 429–447 (1993). https://doi.org/10.1142/9789814354363_0019
54. Agarwal, P.K., Sharir, M., Toledo, S.: Applications of parametric searching in geometric optimization. *Journal of Algorithms* **17**(3), 292–318 (1994). <https://doi.org/10.1006/jagm.1994.1038>
55. Hashizume, S., Fukushima, M., Katoh, N., Ibaraki, T.: Approximation algorithms for combinatorial fractional programming problems. *Mathematical Programming* **37**(3), 255–267 (1987). <https://doi.org/10.1007/BF02591737>
56. Toledo, S.: Approximate parametric searching. *Information Processing Letters* **47**(1), 1–4 (1993). [https://doi.org/10.1016/0020-0190\(93\)90149-4](https://doi.org/10.1016/0020-0190(93)90149-4)
57. Schwartz, J.T., Sharir, M.: On the two-dimensional davenport-schinzel problem. *Journal of Symbolic Computation* **10**(3), 371–393 (1990). [https://doi.org/10.1016/S0747-7171\(08\)80070-3](https://doi.org/10.1016/S0747-7171(08)80070-3)
58. Przybylski, A., Gandibleux, X., Ehrgott, M.: A recursive algorithm for finding all nondominated extreme points in the outcome set of a multiobjective integer programme. *INFORMS Journal on Computing* **22**(3), 371–386 (2009). <https://doi.org/10.1287/ijoc.1090.0342>
59. Agarwal, P.K., Sharir, M.: Davenport–Schinzel sequences and their geometric applications. In: Sack, J.-R., Urrutia, J. (eds.) *Handbook of Computational Geometry*, pp. 1–47. North-Holland, Amsterdam (2000). <https://doi.org/10.1016/B978-044482537-7/50002-4>
60. Fernández-Baca, D.: Space-sweep algorithms for parametric optimization. In: *Proceedings of the 2nd Scandinavian Workshop on Algorithm Theory (SWAT)*, pp. 264–275 (1990). https://doi.org/10.1007/3-540-52846-6_95
61. Waterman, M.S., Eggert, M., Lander, E.: Parametric sequence comparisons. *Proceedings of the National Academy of Sciences of the United States of America* **89**(13), 6090–6093 (1992). <https://doi.org/10.1073/pnas.89.13.6090>
62. Zimmer, R., Lengauer, T.: Fast and numerically stable parametric alignment of biosequences. In: *Proceedings of the 1st Annual International Conference on Computational Molecular Biology (RECOMB)*, pp. 344–353 (1997). <https://doi.org/10.1145/267521.267894>
63. Halfmann, P., Dietz, T., Przybylski, A., Ruzika, S.: An inner approximation method to compute the weight set decomposition of a triobjective mixed-integer problem. *Journal of Global Optimization* **77**(4), 715–742 (2020). <https://doi.org/10.1007/s10898-020-00898-9>
64. Benson, H.P., Sun, E.: A weight set decomposition algorithm for finding all efficient extreme points in the outcome set of a multiple objective linear program. *European Journal of Operational Research* **139**(1), 26–41 (2002). [https://doi.org/10.1016/S0377-2217\(01\)00153-9](https://doi.org/10.1016/S0377-2217(01)00153-9)
65. Halfmann, P.: Advances in multiobjective optimisation: Scalarisation, approximation, and complexity. PhD thesis, Department of Mathematics, Technical University of Kaiserslautern (2021). <https://doi.org/10.13140/RG.2.2.26352.94726>
66. Alves, M.J., Costa, J.P.: Graphical exploration of the weight space in three-objective mixed integer linear programs. *European Journal of Operational Research* **248**(1), 72–83 (2016). <https://doi.org/10.1016/j.ejor.2015.06.072>
67. Fernández-Baca, D., Srinivasan, S.: Constructing the minimization diagram of a two-parameter problem. *Operations Research Letters* **10**(2), 87–93 (1991). [https://doi.org/10.1016/0167-6377\(91\)90092-4](https://doi.org/10.1016/0167-6377(91)90092-4)
68. Heyde, F., Löhne, A.: Geometric duality in multiple objective linear programming. *SIAM Journal on Optimization* **19**(2), 836–845 (2008). <https://doi.org/10.1137/060674831>
69. Fernández-Baca, D., Seppäläinen, T., Slutzki, G.: Parametric multiple sequence alignment and phylogeny construction. *Journal of Discrete Algorithms* **2**(2), 271–287 (2004). [https://doi.org/10.1016/S1570-8667\(03\)00078-9](https://doi.org/10.1016/S1570-8667(03)00078-9)
70. Dobkin, D., Edelsbrunner, H., Yap, C.K.: Probing convex polytopes. In: *Proceedings of the 18th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 424–432 (1986). <https://doi.org/10.1145/12130.12174>

71. Ehrgott, M., Löhne, A., Shao, L.: A dual variant of Benson's "outer approximation algorithm" for multiple objective linear programming. *Journal of Global Optimization* **52**(4), 757–778 (2012). <https://doi.org/10.1007/s10898-011-9709-y>
72. Hamel, A.H., Löhne, A., Rudloff, B.: Benson type algorithms for linear vector optimization and applications. *Journal of Global Optimization* **59**(4), 811–836 (2014). <https://doi.org/10.1007/s10898-013-0098-2>
73. Löhne, A., Weißing, B.: The vector linear program solver Bensolve - notes on theoretical background. *European Journal of Operational Research* **260**(3), 807–813 (2017). <https://doi.org/10.1016/j.ejor.2016.02.039>
74. Bökler, F., Mutzel, P.: Output-sensitive algorithms for enumerating the extreme nondominated points of multiobjective combinatorial optimization problems. In: *Proceedings of the 23rd European Symposium on Algorithms (ESA)*. Lecture Notes in Computer Science, vol. 9294, pp. 288–299 (2015). https://doi.org/10.1007/978-3-662-48350-3_25
75. Borndörfer, R., Schenker, S., Skutella, M., Strunk, T.: PolySCIP. In: *Proceedings of the 5th International Congress on Mathematical Software (ICMS)*, pp. 259–264 (2016). https://doi.org/10.1007/978-3-319-42432-3_32
76. Bökler, F., Nemesch, L., Wagner, M.H.: PaMILO: A solver for multi-objective mixed integer linear optimization and beyond. In: *Operations Research Proceedings 2022*. Lecture Notes in Operations Research, pp. 163–170 (2023). https://doi.org/10.1007/978-3-031-24907-5_20
77. Chazelle, B.: An optimal convex hull algorithm in any fixed dimension. *Discrete & Computational Geometry* **10**(4), 377–409 (1993). <https://doi.org/10.1007/BF02573985>
78. Bökler, F., Parragh, S., Sinnl, M., Tricoire, F.: An outer approximation algorithm for generating the Edgeworth-Pareto hull of multi-objective mixed-integer linear programming problems. *Mathematical Methods of Operations Research* **100**, 263–290 (2024). <https://doi.org/10.1007/s00186-023-00847-8>
79. Benson, H.P.: An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem. *Journal of Global Optimization* **13**(1), 1–24 (1998). <https://doi.org/10.1023/A:1008215702611>
80. Özpınar, Ö., Köksalan, M.: An exact algorithm for finding extreme supported nondominated points of multiobjective mixed integer programs. *Management Science* **56**(12), 2302–2315 (2010). <https://doi.org/10.1287/mnsc.1100.1248>
81. Przybylski, A., Klamroth, K., Lacour, R.: A Simple and Efficient Dichotomic Search Algorithm for Multi-Objective Mixed Integer Linear Programs. Preprint (2019). <https://doi.org/10.48550/arXiv.1911.08937>
82. Katoh, N., Ibaraki, T.: A parametric characterization and an ε -approximation scheme for the minimization of a quasiconcave program. *Discrete Applied Mathematics* **17**(1), 39–66 (1987). [https://doi.org/10.1016/0166-218X\(87\)90006-0](https://doi.org/10.1016/0166-218X(87)90006-0)
83. Murty, K.G.: Computational complexity of parametric linear programming. *Mathematical Programming* **19**(1), 213–219 (1980). <https://doi.org/10.1007/BF01581642>
84. Herceg, M., Kvasnica, M., Jones, C.N., Morari, M.: Multi-parametric toolbox 3.0. In: *Proceedings of the 2013 European Control Conference (ECC)*, pp. 502–510 (2013). <https://doi.org/10.23919/ECC.2013.6669862>
85. Oberdieck, R., Dangelakis, N.A., Papathanasiou, M.M., Nascu, I., Pistikopoulos, E.N.: Pop-parametric optimization toolbox. *Industrial & Engineering Chemistry Research* **55**(33), 8979–8991 (2016). <https://doi.org/10.1021/acs.iecr.6b01913>
86. Carstensen, P.J.: The complexity of some problems in parametric linear and combinatorial programming. PhD thesis, University of Michigan (1983)
87. Mulmuley, K., Shah, P.: A lower bound for the shortest path problem. *Journal of Computer and System Sciences* **63**(2), 253–267 (2001). <https://doi.org/10.1006/jcss.2001.1766>
88. Gajjar, K., Radhakrishnan, J.: Parametric shortest paths in planar graphs. In: *Proceedings of the 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 876–895 (2019). <https://doi.org/10.1109/FOCS.2019.00057>
89. Nikolova, E.: Strategic algorithms. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology (2009)
90. Karp, R.M., Orlin, J.B.: Parametric shortest path algorithms with an application to cyclic staffing. *Discrete Applied Mathematics* **3**(1), 37–45 (1981). [https://doi.org/10.1016/0166-218X\(81\)90026-3](https://doi.org/10.1016/0166-218X(81)90026-3)
91. Young, N.E., Tarjan, R.E., Orlin, J.B.: Faster parametric shortest path and minimum-balance algorithms. *Networks* **21**(2), 205–221 (1991). <https://doi.org/10.1002/net.3230210206>
92. Fredman, M.L., Tarjan, R.E.: Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM* **34**(3), 596–615 (1987). <https://doi.org/10.1145/28869.28874>

93. Séré, B., Jezequel, L., Lime, D.: An algorithm for single-source shortest paths enumeration in parameterized weighted graphs. In: Proceedings of the 15th International Conference on Language and Automata Theory and Applications (LATA). Lecture Notes in Computer Science, vol. 12638, pp. 279–290 (2021). <https://doi.org/10.1007/978-3-030-68195-1>
94. Erickson, J.: Maximum flows and parametric shortest paths in planar graphs. In: Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 794–804 (2010). <https://doi.org/10.1137/1.9781611973075.65>
95. Correa, J., Harks, T., Kreuzen, V.J.C., Matuschke, J.: Fare evasion in transit networks. *Operations Research* **65**(1), 165–183 (2017). <https://doi.org/10.1287/opre.2016.1560>
96. Nikolova, E., Kelner, J.A., Brand, M., Mitzenmacher, M.: Stochastic shortest paths via quasi-convex maximization. In: Proceedings of the 14th European Symposium on Algorithms (ESA). Lecture Notes in Computer Science, vol. 4168, pp. 552–563 (2006). https://doi.org/10.1007/11841036_50
97. Chakraborty, S., Fischer, E., Lachish, O., Yuster, R.: Two-phase algorithms for the parametric shortest path problem. In: Proceedings of the 27th International Symposium on Theoretical Aspects of Computer Science (STACS), pp. 167–178 (2010). <https://doi.org/10.4230/LIPIcs.STACS.2010.2452>
98. Foschini, L., Hershberger, J., Suri, S.: On the complexity of time-dependent shortest paths. *Algorithmica* **68**(4), 1075–1097 (2014). <https://doi.org/10.1007/s00453-012-9714-7>
99. Bökler, F., Chimani, M.: Approximating multiobjective shortest path in practice. In: 2020 Proceedings of the Symposium on Algorithm Engineering and Experiments (ALENEX), pp. 120–133 (2020). <https://doi.org/10.1137/1.9781611976007.10>
100. Ahuja, R.K., Magnanti, T.L., Orlin, J.B.: *Network Flows: Theory, Algorithms, and Applications*, 2nd edn. Pearson, London, UK (1993)
101. Mulmuley, K.: Lower bounds in a parallel model without bit operations. *SIAM Journal on Computing* **28**(4), 1460–1509 (1999). <https://doi.org/10.1137/S0097539794282930>
102. Zadeh, N.: A bad network problem for the simplex method and other minimum cost flow algorithms. *Mathematical Programming* **5**(1), 255–266 (1973). <https://doi.org/10.1007/BF01580132>
103. Allman, M., Lo, V., McCormick, S.T.: Complexity of source-sink monotone 2-parameter min cut. *Operations Research Letters* **50**(1), 84–90 (2022). <https://doi.org/10.1016/j.orl.2021.12.009>
104. Kolmogorov, V., Boykov, Y., Rother, C.: Applications of parametric maxflow in computer vision. In: Proceedings of the 11th International Conference on Computer Vision (ICCV), pp. 1–8 (2007). <https://doi.org/10.1109/ICCV.2007.4408910>
105. Asad, M.W.A., Dimitrakopoulos, R.: Implementing a parametric maximum flow algorithm for optimal open pit mine design under uncertain supply and demand. *Journal of the Operational Research Society* **64**(2), 185–197 (2013). <https://doi.org/10.1057/jors.2012.26>
106. Sangeorzan, L., Parpalea, M., Parpalea, M.M., Repanovici, A., Matefi, R., Nicolae, I.: A preflow approach to maximum flows in parametric networks applied to assessing the legal information. *International Journal of Computers, Communications and Control* **11**(1), 113–125 (2016). <https://doi.org/10.15837/ijccc.2016.1.1440>
107. Stone, H.S.: Critical load factors in two-processor distributed systems. *IEEE Transactions on Software Engineering* **SE-4**(3), 254–258 (1978). <https://doi.org/10.1109/TSE.1978.231502>
108. Gallo, G., Grigoriadis, M., Tarjan, R.E.: A fast parametric maximum flow algorithm and applications. *SIAM Journal on Computing* **18**(1), 30–55 (1989). <https://doi.org/10.1137/0218003>
109. Goldberg, A.V., Tarjan, R.E.: A new approach to the maximum-flow problem. *Journal of the ACM* **35**(4), 921–940 (1988). <https://doi.org/10.1145/48014.61051>
110. Arai, T., Ueno, S., Kajitani, Y.: Generalization of a theorem on the parametric maximum flow problem. *Discrete Applied Mathematics* **41**(1), 69–74 (1993). [https://doi.org/10.1016/0166-218X\(93\)90245-J](https://doi.org/10.1016/0166-218X(93)90245-J)
111. McCormick, S.T.: Fast algorithms for parametric scheduling come from extensions to parametric maximum flow. *Operations Research* **47**(5), 744–756 (1999). <https://doi.org/10.1287/opre.47.5.744>
112. Gusfield, D., Tardos, É.: A faster parametric minimum-cut algorithm. *Algorithmica* **11**(3), 278–290 (1994). <https://doi.org/10.1007/BF01240737>
113. Hochbaum, D.S.: The pseudoflow algorithm: A new algorithm for the maximum-flow problem. *Operations Research* **56**(4), 992–1009 (2008). <https://doi.org/10.1287/opre.1080.0524>
114. Tarjan, R.E., Ward, J., Zhang, B., Zhou, Y., Mao, J.: Balancing applied to maximum network flow problems. In: Proceedings of the 14th European Symposium on Algorithms (ESA). Lecture Notes in Computer Science, vol. 4168, pp. 612–623 (2006). https://doi.org/10.1007/11841036_55
115. Brumelle, S., Granot, D., Liu, L.: Ordered optimal solutions and parametric minimum cut problems. *Discrete Optimization* **2**(2), 123–134 (2005). <https://doi.org/10.1016/j.disopt.2005.03.002>
116. Scutellà, M.G.: A note on the parametric maximum flow problem and some related reoptimization issues. *Annals of Operations Research* **150**, 231–244 (2007). <https://doi.org/10.1007/s10479-006-0155-z>

117. Granot, F., McCormick, S.T., Queyranne, M., Tardella, F.: Structural and algorithmic properties for parametric minimum cuts. *Mathematical Programming* **135**(1), 337–367 (2012). <https://doi.org/10.1007/s10107-011-0463-1>
118. McCormick, S.T., Ervolina, T.R.: Computing maximum mean cuts. *Discrete Applied Mathematics* **52**(1), 53–70 (1994). [https://doi.org/10.1016/0166-218X\(92\)00188-R](https://doi.org/10.1016/0166-218X(92)00188-R)
119. Radzik, T.: Minimizing capacity violations in a transshipment network. In: *Proceedings of the 3rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 185–194 (1992)
120. Radzik, T.: Newton's method for fractional combinatorial optimization. In: *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 659–669 (1992). <https://doi.org/10.1109/SFCS.1992.267785>
121. Radzik, T.: Parametric flows, weighted means of cuts, and fractional combinatorial optimization. In: Pardalos, P.M. (ed.) *Complexity in Numerical Optimization*, pp. 351–386. World Scientific, Singapore (1993). https://doi.org/10.1142/9789814354363_0016
122. Radzik, T.: Fractional combinatorial optimization. In: Du, D., Pardalos, P.M. (eds.) *Handbook of Combinatorial Optimization* vol. 1, pp. 429–478. Springer, New York (1998). https://doi.org/10.1007/978-1-4613-0303-9_6
123. Aissi, H., Mahjoub, A.R., McCormick, S.T., Queyranne, M.: Strongly polynomial bounds for multiobjective and parametric global minimum cuts in graphs and hypergraphs. *Mathematical Programming* **154**(1), 3–28 (2015). <https://doi.org/10.1007/s10107-015-0944-8>
124. Karger, D.R.: Enumerating parametric global minimum cuts by random interleaving. In: *Proceedings of the 48th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 542–555 (2016). <https://doi.org/10.1145/2897518.2897578>
125. Hamacher, H.W., Pedersen, C.R., Ruzika, S.: Multiple objective minimum cost flow problems: A review. *European Journal of Operational Research* **176**(3), 1404–1422 (2007). <https://doi.org/10.1016/j.ejor.2005.09.033>
126. Pulat, P.S., Huarng, F., Lee, H.: Efficient solutions for the bicriteria network flow problem. *Computers & Operations Research* **19**(7), 649–655 (1992). [https://doi.org/10.1016/0305-0548\(92\)90034-3](https://doi.org/10.1016/0305-0548(92)90034-3)
127. Lee, H., Simin Pulat, P.: Bicriteria network flow problems: Continuous case. *European Journal of Operational Research* **51**(1), 119–126 (1991). [https://doi.org/10.1016/0377-2217\(91\)90151-K](https://doi.org/10.1016/0377-2217(91)90151-K)
128. Sedeño-Noda, A., González-Martín, C.: The biobjective minimum cost flow problem. *European Journal of Operational Research* **124**(3), 591–600 (2000). [https://doi.org/10.1016/S0377-2217\(99\)00191-5](https://doi.org/10.1016/S0377-2217(99)00191-5)
129. Sedeño-Noda, A., González-Martín, C.: An alternative method to solve the biobjective minimum cost flow problem. *Asia-Pacific Journal of Operational Research* **20**(2), 241–260 (2003)
130. Eusébio, A., Figueira, J.R., Ehrgott, M.: A primal–dual simplex algorithm for bi-objective network flow problems. *4OR* **7**(3), 255–273 (2009) <https://doi.org/10.1007/s10288-008-0087-3>
131. Raith, A., Sedeño-Noda, A.: Finding extreme supported solutions of biobjective network flow problems: An enhanced parametric programming approach. *Computers & Operations Research* **82**, 153–166 (2017). <https://doi.org/10.1016/j.cor.2017.01.004>
132. Raith, A., Ehrgott, M.: A two-phase algorithm for the biobjective integer minimum cost flow problem. *Computers & Operations Research* **36**(6), 1945–1954 (2009). <https://doi.org/10.1016/j.cor.2008.06.008>
133. Sangeorzan, L., Parpalea, M., Parpalea, M.M.: Partitioning preflow-pull algorithm for the parametric network flow problem: a linguistic rule-based constraints optimisation approach. In: *Proceedings of the 11th WSEAS International Conference on Neural Networks*, pp. 111–116 (2010)
134. Ciurea, E., Parpalea, M.: Balancing algorithm for the minimum flow problem in parametric bipartite networks. In: *Proceedings of the 14th WSEAS International Conference on Computers*, pp. 226–231 (2010)
135. Ciurea, E., Parpalea, M.: A sequential algorithm for finding the solution of the parametric minimum flow problem. *Carpathian Journal of Mathematics* **28**(1), 47–58 (2012)
136. Ciurea, E., Parpalea, M.: Shortest conditional decreasing path algorithm for the parametric minimum flow problem. *Bulletin Mathématique de la Société des Sciences Mathématiques de Roumanie* **56(104)**(4), 387–401 (2013)
137. Parpalea, M., Ciurea, E.: Minimum parametric flow - a partitioning approach. *British Journal of Applied Science & Technology* **13**(6), 1–8 (2016). <https://doi.org/10.9734/BJAST/2016/22636>
138. Parpalea, M.: Parametric flows in static networks. *Bulletin of the Transilvania University of Brasov* **12(61)**(1), 157–176 (2019) <https://doi.org/10.31926/but.mif.2019.12.61.1.13>
139. Gusfield, D.: Bounds for the parametric minimum spanning tree problem. In: *Proceedings of the West Coast Conference on Combinatorics, Graph Theory and Computing*, pp. 173–181 (1980)
140. Eppstein, D.: A stronger lower bound on parametric minimum spanning trees. *Algorithmica* **85**(6), 1738–1753 (2023). <https://doi.org/10.1007/s00453-022-01024-9>

141. Dey, T.K.: Improved bounds for planar k -sets and related problems. *Discrete & Computational Geometry* **19**(3), 373–382 (1998). <https://doi.org/10.1007/PL00009354>
142. Erdős, P., Lovász, L.M., Simmons, A., Straus, E.G.: Dissection graphs of planar point sets. In: Srivastava, J.N. (ed.) *A Survey of Combinatorial Theory*, pp. 139–149 (1973). Chap. 13. <https://doi.org/10.1016/B978-0-7204-2262-7.50018-1>
143. Lovász, L.M.: On the number of halving lines. *Annales Universitatis Scientiarum Budapestinensis de Rolando Eötvös Nominatae, Sect. Math.* **14**, 107–108 (1971)
144. Pach, J., Steiger, W., Szemerédi, E.: An upper bound on the number of planar k -sets. *Discrete & Computational Geometry* **7**, 109–123 (1992). <https://doi.org/10.1007/BF02187829>
145. Seipp, F.: On adjacency, cardinality, and partial dominance in discrete multiple objective optimization. PhD thesis, Department of Mathematics, Technical University of Kaiserslautern (2013)
146. Hassin, R., Tamir, A.: Maximizing classes of two-parameter objectives over matroids. *Mathematics of Operations Research* **14**(2), 362–375 (1989). <https://doi.org/10.1287/moor.14.2.362>
147. Ishii, H., Shiode, S., Nishida, T., Namasuya, Y.: Stochastic spanning tree problem. *Discrete Applied Mathematics* **3**(4), 263–273 (1981). [https://doi.org/10.1016/0166-218X\(81\)90004-4](https://doi.org/10.1016/0166-218X(81)90004-4)
148. Agarwal, P.K., Eppstein, D., Guibas, L.J., Henzinger, M.R.: Parametric and kinetic minimum spanning trees. In: *Proceedings of the 39th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 596–605 (1998). <https://doi.org/10.1109/SFCS.1998.743510>
149. Fernández-Baca, D., Slutzki, G.: Linear-time algorithms for parametric minimum spanning tree problems on planar graphs. *Theoretical Computer Science* **181**(1), 57–74 (1997). [https://doi.org/10.1016/S0304-3975\(96\)00262-9](https://doi.org/10.1016/S0304-3975(96)00262-9)
150. Fernández-Baca, D.: Multi-parameter minimum spanning trees. In: *Proceedings of the 4th Latin American Symposium on Theoretical Informatics (LATIN)*, pp. 217–226 (2000). https://doi.org/10.1007/10719839_22
151. Katoh, N., Tokuyama, T.: Notes on computing peaks in k -levels and parametric spanning trees. In: *Proceedings of the 17th Annual Symposium on Computational Geometry (SoCG)*, pp. 241–248 (2001). <https://doi.org/10.1145/378583.378675>
152. Chan, T.M.: Finding the shortest bottleneck edge in a parametric minimum spanning tree. In: *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 917–918 (2005)
153. Hausbrandt, N., Bachtler, O., Ruzika, S., Schäfer, L.E.: Parametric matroid interdiction. *Discrete Optimization* **51**, 1572–1586 (2024). <https://doi.org/10.1016/j.disopt.2024.100823>
154. Eben-Chaïme, M.: Parametric solution for linear bicriteria knapsack models. *Management Science* **42**(11), 1565–1575 (1996). <https://doi.org/10.1287/mnsc.42.11.1565>
155. Burkard, R.E., Pferschy, U.: The inverse-parametric knapsack problem. *European Journal of Operational Research* **83**(2), 376–393 (1995). [https://doi.org/10.1016/0377-2217\(95\)00014-H](https://doi.org/10.1016/0377-2217(95)00014-H)
156. Erlebach, T., Kellerer, H., Pferschy, U.: Approximating multiobjective knapsack problems. *Management Science* **48**(12), 1603–1612 (2002). <https://doi.org/10.1287/mnsc.48.12.1603.445>
157. Lawler, E.: *Combinatorial Optimization: Networks and Matroids*. Dover Books on Mathematics Series. Dover Publications, Mineola, NY (2001)
158. Hoffman, A.J., Markowitz, H.M.: A note on shortest path, assignment, and transportation problems. *Naval Research Logistics Quarterly* **10**(1), 375–379 (1963). <https://doi.org/10.1002/nav.3800100132>
159. Rosenmann, A.: Computing the sequence of k -cardinality assignments. *Journal of Combinatorial Optimization* **44**(2), 1265–1283 (2022). <https://doi.org/10.1007/s10878-022-00889-4>
160. Weber, G.M.: Sensitivity analysis of optimal matchings. *Networks* **11**(1), 41–56 (1981). <https://doi.org/10.1002/net.3230110105>
161. Gusfield, D., Irving, R.W.: Parametric stable marriage and minimum cuts. *Information Processing Letters* **30**(5), 255–259 (1989). [https://doi.org/10.1016/0020-0190\(89\)90204-4](https://doi.org/10.1016/0020-0190(89)90204-4)
162. Hrubeš, P., Yehudayoff, A.: Shadows of newton polytopes. *Israel Journal of Mathematics* **256**(1), 311–343 (2023). <https://doi.org/10.1007/s11856-023-2510-z>
163. Lin, M., Jaillet, P.: On the quickest flow problem in dynamic networks - a parametric min-cost flow approach. In: *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 1343–1356 (2015). <https://doi.org/10.1137/1.9781611973730.89>