Information Processing Letters 120 (2017) 11–15

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Approximation schemes for the parametric knapsack problem

Alberto Giudici [a], Pascal Halffmann [b,1], Stefan Ruzika [b,2], Clemens Thielen [c,*]

[a] Rotterdam School of Management, Erasmus University Rotterdam, 3000 DR Rotterdam, The Netherlands
[b] Mathematical Institute, University of Koblenz-Landau, Campus Koblenz, D-56070 Koblenz, Germany
[c] Department of Mathematics, University of Kaiserslautern, Paul-Ehrlich-Str. 14, D-67663 Kaiserslautern, Germany

ARTICLE INFO

Article history:
Received 1 September 2016
Received in revised form 13 December 2016
Accepted 13 December 2016
Available online 15 December 2016
Communicated by B. Doerr

Keywords:
Parametric optimization problems
Approximation algorithms
Bicriteria optimization problems

ABSTRACT

We consider the (linear) parametric 0–1 knapsack problem in which the profits of the items are affine-linear functions of a real-valued parameter and the task is to compute a solution for all values of the parameter. For this problem, it is known that the piecewise linear convex function mapping the parameter to the optimal objective value of the corresponding instance (called the *optimal value function*) can have exponentially many breakpoints (points of slope change), which implies that every optimal algorithm for the problem must output a number of solutions that is exponential in the number of items.
We provide the first (parametric) polynomial time approximation scheme (PTAS) for the parametric 0–1 knapsack problem. Moreover, we exploit the connection between the parametric problem and the bicriteria problem in order to show that the parametric 0–1 knapsack problem admits a parametric FPTAS when the parameter is restricted to the positive real line and the slopes and intercepts of the affine-linear profit functions of the items are nonnegative. The method used to obtain this result applies to many linear parametric optimization problems and provides a general connection between bicriteria and linear parametric optimization problems.

1. Introduction

The knapsack problem is a well-studied combinatorial optimization problem with numerous applications. Given a knapsack capacity and a set of n items with different weights and profits, the task in the classical 0–1 knapsack problem is to select a subset of the items with maximum total profit subject to the constraint that the total weight of the selected items may not exceed the knapsack capacity. The 0–1 knapsack problem is NP-hard, but it admits a fully polynomial time approximation scheme (FPTAS) and can be solved exactly in pseudo-polynomial time by dynamic programming (cf. [1]).

The (linear) parametric 0–1 knapsack problem is a generalization of the 0–1 knapsack problem in which the profits of the items are affine-linear functions of a parameter $\lambda \in \mathbb{R}$. Here, the profit of each item i is given as $p_i = p_i(\lambda) = a_i + \lambda b_i$ with $a_i, b_i \in \mathbb{Z}$ and the problem can be written as

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} (a_i + \lambda b_i) \cdot x_i \\ \text{s.t.} \quad & \sum_{i=1}^{n} w_i \cdot x_i \le B \\ & x_i \in \{0, 1\} \qquad \text{for } i = 1, \dots, n \end{aligned}$$

* Corresponding author. Fax: +49 631 205 4737.
E-mail addresses: giudici@rsm.nl (A. Giudici), halffmann@uni-koblenz.de (P. Halffmann), ruzika@uni-koblenz.de (S. Ruzika), thielen@mathematik.uni-kl.de (C. Thielen).

1 The research of Pascal Halffmann was supported by DFG grant RU 1524/4-1.
2 Stefan Ruzika acknowledges support by BMBF grant 13N12825 and DAAD project 57128839.

http://dx.doi.org/10.1016/j.ipl.2016.12.003

where $B \in \mathbb{N}$ denotes the knapsack capacity and $w_i \in \mathbb{N}_{>0}$ denotes the weight of item i. We note that, since positive as well as negative values a_i, b_i are allowed, some items might have negative profits even for positive values of λ. Moreover, the profits of some items might increase with increasing λ while the profits of other items might decrease.

For linear parametric optimization problems such as the parametric 0–1 knapsack problem, one is interested in obtaining optimal solutions of the problem for all values of λ on the real line (or within a given interval). Since the objective values of feasible solutions are affine-linear functions of λ, it is easy to see that such a collection of optimal solutions is given by a finite, increasing sequence of parameter values $-\infty = \lambda_0, \ldots, \lambda_{K+1} = +\infty$ together with an optimal solution for each interval $(-\infty, \lambda_1], [\lambda_1, \lambda_2], \ldots, [\lambda_{K-1}, \lambda_K], [\lambda_K, +\infty)$, where an *optimal* solution for an interval is a feasible solution that is optimal for all values of λ within the interval. The function mapping $\lambda \in \mathbb{R}$ to the optimal objective value of the given instance for this value of λ is called the *optimal value function* (or the *optimal cost curve*). The above structure of optimal solutions implies that the optimal value function is piecewise linear and convex (concave in case of a minimization problem) and its *breakpoints* (points of slope change) are exactly at the points $\lambda_1, \ldots, \lambda_K$ (assuming that K was chosen as small as possible). Thus, the number K of breakpoints is a natural measure of the complexity of the problem. For the parametric 0–1 knapsack problem, Carstensen [2] showed that the number of breakpoints can be exponential in the number n of items, which implies that every optimal algorithm for the parametric 0–1 knapsack problem has to output an exponential number of solutions. Moreover, she raised the question which approximations of the optimal value function are obtainable in polynomial time.

1.1. Previous work

Linear parametric optimization problems in which the objective values of feasible solutions are affine-linear functions of a real parameter are widely studied in the literature. Besides the parametric 0–1 knapsack problem studied here, examples include the parametric shortest path problem [3–6], the parametric minimum spanning tree problem [7], and the parametric minimum cost flow problem [2]. While the number of breakpoints in the optimal value function of the parametric minimum spanning tree problem is known to be polynomial in the input size of the problem [7], the optimal value function of the parametric minimum cost flow problem can have exponentially many breakpoints even when the slopes of the affine-linear functions are restricted to the set $\{0, 1\}$ [2] and the optimal value function of the parametric shortest path problem can have pseudo-exponentially many breakpoints ($n^{\Omega(\log n)}$ on graphs with n nodes) [5,6]. When the slopes of the affine-linear functions are integers in $\{-M, \ldots, M\}$ for some constant $M \in \mathbb{N}$, however, the number of breakpoints in the optimal value function of the parametric shortest path problem becomes polynomial [4]. In several variants of parametric maximum flow problems, it is known that the minimum cuts satisfy so-called "nesting properties", which imply that there are at most $n-1$ breakpoints in the optimal value function on graphs with n nodes [8–11]. Parametric versions of general linear programs, mixed integer programs, and nonlinear programs (where the most general cases consider also non-affine dependence on the parameter as well as constraints depending on the parameter) are also widely studied. For an extensive literature review on these problems, we refer to [12].

The parametric 0–1 knapsack problem first appeared in the work of Carstensen [2], who shows that the number of breakpoints in the optimal value function can be exponential in the number of items. This holds even when restricting λ to a compact interval on the positive real line $\mathbb{R}_{>0}$ with the property that all profits are positive within this interval. However, she also shows that the number of breakpoints in the optimal value function of any linear parametric binary integer program becomes linear in the number of variables when the slopes and/or intercepts of the affine-linear functions are integers in $\{-M, \ldots, M\}$ for some constant $M \in \mathbb{N}$. In particular, this implies that the number of breakpoints in the optimal value function of the parametric 0–1 knapsack problem becomes linear in the number of items under this assumption. Eben-Chaime [13] shows that the optimal value function of the parametric 0–1 knapsack problem (together with a corresponding optimal solution between any two breakpoints) can be computed in $\mathcal{O}(KnB)$, where K denotes the number of breakpoints. This is achieved by using a general method of Eisner and Severance [14], which can be used to solve any instance of a linear parametric optimization problem with K breakpoints in the optimal value function by solving the instance for $\mathcal{O}(K)$ fixed values of the parameter.

A problem closely related to the parametric 0–1 knapsack problem is the *inverse-parametric knapsack problem* [15], which consists of computing the smallest value of λ for which the optimal value function of the parametric 0–1 knapsack problem has value equal to a prespecified solution value. For this problem, pseudo-polynomial (exact) algorithms are provided by Burkard and Pferschy [15].

The parametric 0–1 knapsack problem is also closely related to the bicriteria 0–1 knapsack problem since it can be interpreted as the weighted sum scalarization of the bicriteria problem. Thus, the optimal solutions of the parametric problem on the positive real line are exactly the supported efficient solutions of the bicriteria problem. For the bicriteria and multicriteria 0–1 knapsack problem, where the profit of each item in all objective functions is assumed to be nonnegative, several (multicriteria) FPTAS are known [16–19], i.e., algorithms that, given $\epsilon > 0$, compute in time polynomial in the size of the input and $1/\epsilon$ a set of solutions that, for each efficient solution, contains a solution that is at most at a factor $(1-\epsilon)$ worse in all objective functions.

1.2. Our contribution

We show that the parametric 0–1 knapsack problem admits a (parametric) polynomial time approximation scheme (PTAS). This means that, for any given $\epsilon > 0$, there exists an algorithm with running time polynomial in the input size (but possibly exponential in $1/\epsilon$) that computes a partition of the real line into polynomially many intervals together with a feasible solution for each interval such that this solution is a $(1-\epsilon)$-approximation for the given problem instance for all values of λ within the interval.

Moreover, we exploit the connection between the parametric problem and the bicriteria problem in order to show that the parametric 0–1 knapsack problem admits a parametric FPTAS (i.e., a parametric PTAS whose running time is additionally polynomial in $1/\epsilon$) when $a_i, b_i \geq 0$ for all i (so, in particular, all profits are nondecreasing functions of λ) and λ is restricted to the positive real line $\mathbb{R}_{>0}$.[3] The method we use for obtaining this parametric FPTAS can be used to construct a parametric FPTAS on the positive real line for every linear parametric problem for which the bicriteria version admits a (bicriteria) FPTAS, provided that the slopes and intercepts of the affine functions are all nonnegative.

2. Obtaining a parametric PTAS

In this section, we present our parametric PTAS for the parametric 0–1 knapsack problem. The parametric PTAS is based on the following classical PTAS for the (non-parametric) 0–1 knapsack problem due to Sahni [20]:

Algorithm 1. Non-parametric 0–1 knapsack PTAS.

Input: An instance of the 0–1 knapsack problem given by the knapsack capacity B, the weights $w_1, \ldots, w_n$, and the profits $p_1, \ldots, p_n$.

Output: A feasible packing obtaining profit at least $(1-\epsilon)$ times the optimal profit.

1 Choose $k := \min\{\frac{1}{\epsilon} - 1, n\}$
2 **for** each $L \subseteq \{1, \ldots, n\}$ with $|L| \leq k$ **do**
3   **if** $\sum_{j \in L} w_j \leq B$ **then**
4   Set $S_L := L$
5   **for** all items $i \in \{1, \ldots, n\} \setminus L$ in nonincreasing order of p_i/w_i **do**
6   **if** $\sum_{j \in S_L \cup \{i\}} w_j \leq B$ and $p_i \geq 0$ **then**
7         $S_L := S_L \cup \{i\}$
8   **end if**
9   **end for**
10   **end if**
11 **end for**
12 **return** Best solution S_L obtained.

Note that we added the condition $p_i \geq 0$ in line 6 of Algorithm 1 since we do not assume the profits to be nonnegative. Also note that, in the non-parametric problem, it is possible to consider only items $i \in \{1, \ldots, n\} \setminus L$ with profit less than or equal to the smallest profit of an item in L in line 5 of Algorithm 1. However, the original version of the algorithm stated here is more suitable for the construction of our parametric PTAS, so we stick to this version here.

[3] Rewriting the profits $p_i = a_i + \lambda b_i$ as $p_i = a_i + (-\lambda) \cdot (-b_i)$ shows that the same technique can be used when $a_i \geq 0$, $b_i \leq 0$ for all i and λ is restricted to the negative real line $\mathbb{R}_{<0}$.

Our parametric PTAS is based on the observation that the solution S_L produced in Algorithm 1 for each specific set L does not change as long as the ordering of the items in $\{1, \ldots, n\} \setminus L$ by p_i/w_i does not change and none of the profits p_i of these items change their sign (which is equivalent to none of the profit densities p_i/w_i changing their sign since w_i is independent of λ). Hence, within any interval I in which none of the n affine functions of λ describing the profit densities of the n items intersect and none of these functions change their sign, the algorithm compares the same solutions S_L in line 12. Obviously, as the profits of the compared solutions S_L are also affine functions of λ, the solution chosen in line 12 may change within the interval I, so we will additionally have to compute the maximum function of these affine functions over I. If we do so and partition I so that we always output the best of the solutions in each subinterval, we obtain an approximation guarantee of $1-\epsilon$ on the whole interval I since the solution we choose for each specific $\lambda \in I$ is identical to the solution Algorithm 1 chooses when applied with weights $p_i = a_i + \lambda b_i$.

In order to implement this idea and analyze the running time, we subdivide the procedure into three steps:

I We first compute the t intersection points of the affine functions $f_i(\lambda) := \frac{p_i(\lambda)}{w_i} = \frac{a_i}{w_i} + \lambda \cdot \frac{b_i}{w_i}$, $i = 1, \ldots, n$, and the zero function $f_0 \equiv 0$ (where $t \leq \frac{n(n+1)}{2} \in \mathcal{O}(n^2)$ is the number of non-parallel pairs of functions among $f_0, \ldots, f_n$) and store them sorted by their abscissae. The abscissae of the t intersection points partition the real line into (at most) $t+1 \in \mathcal{O}(n^2)$ intervals $I_0, \ldots, I_t$ such that none of the functions $f_1, \ldots, f_n$ intersect or change sign within the interior of any interval I_j. Using the optimal line segment intersection algorithm from [21], this step can be performed in $\mathcal{O}(n \log n + t)$ time.[4]

II For each interval I_l, we then apply Algorithm 1 (except for the comparison of the solutions S_L performed in line 12) once with profits $p_i = a_i + \lambda_l b_i$ for some arbitrary λ_l in the interior of I_l (by construction of the intervals I_l, the produced collection of solutions S_L is independent of the choice of λ_l inside I_l). Since the for loop in lines 2–11 iterates over $\sum_{j=0}^{k} \binom{n}{j} \in \mathcal{O}(k \cdot n^k)$ subsets L and all operations performed inside the for-loop can be performed in linear time $O(n)$ after an initial sorting of all items by nonincreasing profit densities p_i/w_i, this step can be performed in $\mathcal{O}(k \cdot n^{k+1})$ time for each interval I_l, which yields a total time requirement of $\mathcal{O}(t \cdot k \cdot n^{k+1})$ for all $t+1$ intervals together.

III For each interval I_l, we then compute the maximum function (within I_l) of the $\mathcal{O}(k \cdot n^k)$ affine functions

[4] Even though the algorithm from [21] is designed for computing the intersection points of finite line segments, it can be applied in our case of affine functions by first computing the intersection points with the smallest and largest abscissa (which can be achieved in $\mathcal{O}(n \log n)$ time by sorting the functions by their slopes and then computing the intersection points of all adjacent pairs of functions in the resulting ordering) and then restricting the functions to line segments on the compact interval between these two abscissae.

of λ given by the profits of the solutions S_L obtained in this interval in the previous step. This yields a subdivision of I_l into several subintervals together with the best solution among the considered solutions S_L on each subinterval (this is the solution we choose for all values of λ within this subinterval). Since the maximum function of a set of m affine functions can be computed in $\mathcal{O}(m \log m)$ time by standard methods from computational geometry (cf. [22]), this step can be performed in $\mathcal{O}(k \cdot n^k \cdot \log(k \cdot n^k)) = \mathcal{O}(k^2 \cdot n^k \cdot \log n)$ time per interval I_l, which yields a total time requirement of $\mathcal{O}(t \cdot k^2 \cdot n^k \cdot \log n)$ for all intervals $I_0, \ldots, I_t$ together.

By using the above time requirements of the three steps, we obtain an overall running time of $\mathcal{O}(t \cdot (k \cdot n^{k+1} + k^2 \cdot n^k \cdot \log n)) = \mathcal{O}(t \cdot k^2 \cdot n^{k+1})$. Hence, we have shown:

Theorem 2. *There exists a parametric PTAS for the parametric 0–1 knapsack problem with running time* $\mathcal{O}(t \cdot \frac{1}{\epsilon^2} \cdot n^{\frac{1}{\epsilon}})$*, where* $t \le \frac{n(n+1)}{2} \in \mathcal{O}(n^2)$ *is upper bounded by the number of pairs of items with different slopes* $\frac{b_i}{w_i}$ *of the profit density function* $\frac{p_i(\lambda)}{w_i}$ *plus the number of items with nonzero* b_i*.*

We remark that, in order to sort all items by nonincreasing profit densities p_i/w_i within each of the intervals $I_0, \ldots, I_t$ in Step II, it is not necessary to compute the ordering from scratch inside each interval. Instead, one can sort the items once by nonincreasing profit densities inside the leftmost interval and then iteratively update this ordering after each interval boundary. Using a result from [23] for the update, this yields a total time requirement of $\mathcal{O}(n \log n + t \cdot n)$ for the sorting in all intervals $I_0, \ldots, I_t$.

3. Relation to the bicriteria problem

As already noted, the optimal solutions of the parametric problem on the positive real line $\mathbb{R}_{>0}$ are exactly the supported efficient solutions of the bicriteria problem (in which the profit vector of item i is (a_i, b_i)). Hence, the Pareto curve of the bicriteria problem contains all optimal solutions of the parametric problem on $\mathbb{R}_{>0}$. The following proposition shows that there is a similar relation between *approximate* Pareto curves and *approximate* solutions of the parametric problem on $\mathbb{R}_{>0}$ when $a_i, b_i \ge 0$ for all i:

Proposition 3. *Let* $P_{\alpha,\beta}$ *denote an* (α,β)*-approximate Pareto curve for the instance of the bicriteria 0–1 knapsack problem with weights* (a_i, b_i) *for each item i and assume that* $a_i, b_i \ge 0$ *for all i. For a fixed* $\lambda > 0$*, let* $x(\lambda) \in P_{\alpha,\beta}$ *be a solution maximizing* $\sum_{i=1}^n (a_i + \lambda b_i) \cdot x_i$ *over all* $x \in P_{\alpha,\beta}$*. Then* $x(\lambda)$ *is a* $\max\{\alpha,\beta\}$*-approximation for the parametric problem at* λ*.*

Proof. Let x^* denote an optimal solution for the parametric problem at λ. Then, since $P_{\alpha,\beta}$ is an (α,β)-approximate Pareto curve, there exists $\bar{x} \in P_{\alpha,\beta}$ such that $\alpha \cdot (\sum_{i=1}^n a_i \cdot \bar{x}_i) \ge \sum_{i=1}^n a_i \cdot x_i^*$ and $\beta \cdot (\sum_{i=1}^n b_i \cdot \bar{x}_i) \ge \sum_{i=1}^n b_i \cdot x_i^*$, which yields

$$\max\{\alpha,\beta\} \cdot \left(\sum_{i=1}^n (a_i + \lambda b_i) \cdot \bar{x}_i\right) \ge \sum_{i=1}^n (a_i + \lambda b_i) \cdot x_i^*.$$

Since $\sum_{i=1}^n (a_i + \lambda b_i) \cdot x(\lambda)_i \ge \sum_{i=1}^n (a_i + \lambda b_i) \cdot \bar{x}_i$, this shows the claim. □

When $a_i, b_i \ge 0$ for all i, Proposition 3 shows that any (α,β)-approximate Pareto curve $P_{\alpha,\beta}$ for an instance of the bicriteria problem contains a $\max\{\alpha,\beta\}$-approximate solution for the corresponding instance of the parametric problem at each positive λ. Note that the assumptions that $a_i, b_i \ge 0$ for all i and $\lambda > 0$ are crucial in Proposition 3 since no similar connection between approximate Pareto curves and approximate solutions of the parametric problem exists otherwise.

Proposition 3 immediately yields a method for obtaining a $\max\{\alpha,\beta\}$-approximation for the parametric problem on $\mathbb{R}_{>0}$ from an (α,β)-approximate Pareto curve $P_{\alpha,\beta}$ if $a_i, b_i \ge 0$ for all i: For each point $x \in P_{\alpha,\beta}$, let g_x denote the affine function mapping $\lambda \in \mathbb{R}_{>0}$ to the objective value (profit) of x in the parametric problem at λ. Then, by Proposition 3, computing the maximum function of the set of affine functions $\{g_x : x \in P_{\alpha,\beta}\}$ over $\mathbb{R}_{>0}$ yields the desired $\max\{\alpha,\beta\}$-approximation for the parametric problem on $\mathbb{R}_{>0}$. Denoting the number of points in $P_{\alpha,\beta}$ by $|P_{\alpha,\beta}|$, this maximum function can be computed in $\mathcal{O}(|P_{\alpha,\beta}| \cdot \log|P_{\alpha,\beta}|)$ time (cf. [22]), so we obtain:

Proposition 4. *If* $a_i, b_i \ge 0$ *for all i, a* $\max\{\alpha,\beta\}$*-approximation for the parametric 0–1 knapsack problem on* $\mathbb{R}_{>0}$ *can be computed from an* (α,β)*-approximate Pareto curve* $P_{\alpha,\beta}$ *for the corresponding instance of the bicriteria problem in* $\mathcal{O}(|P_{\alpha,\beta}| \cdot \log|P_{\alpha,\beta}|)$ *time, where* $|P_{\alpha,\beta}|$ *denotes the number of points in* $P_{\alpha,\beta}$*.*

Proposition 4 and the existence of a (bicriteria) FPTAS for the bicriteria 0–1 knapsack problem directly yield the following corollary:

Corollary 5. *If* $a_i, b_i \ge 0$ *for all i, the parametric 0–1 knapsack problem admits a parametric FPTAS on the positive real line* $\mathbb{R}_{>0}$*.*

Note that Propositions 3 and 4 hold analogously for every linear parametric optimization problem as long as the slopes and intercepts of the affine functions are all nonnegative.

Also note that, since the function $\sum_{i=1}^n (a_i + \lambda b_i) \cdot x_i$ considered in Proposition 3 is linear in $x \in P_{\alpha,\beta}$, there always exists a vertex of the convex hull of $P_{\alpha,\beta}$ at which the maximum of this function over $P_{\alpha,\beta}$ is attained. Thus, Proposition 3 shows that also the (potentially smaller) set of vertices of the convex hull of $P_{\alpha,\beta}$ contains a $\max\{\alpha,\beta\}$-approximate solution for the parametric problem at each positive λ. However, since computing the vertices of the convex hull of $P_{\alpha,\beta}$ already requires $\Omega(|P_{\alpha,\beta}| \cdot \log|P_{\alpha,\beta}|)$ time (cf. [24]), using the set of vertices of the convex hull of $P_{\alpha,\beta}$ instead of $P_{\alpha,\beta}$ in our argumentation would not yield an improved running time in Proposition 4 (as long as the vertices of the convex hull are not already known a priori).

In order to analyze the running time that can be achieved for the parametric FPTAS in Corollary 5, we note that the bicriteria FPTAS for the bicriteria 0–1 knapsack problem from [18] runs in $\mathcal{O}(n^3 \cdot \frac{1}{\epsilon^2} \cdot \log^2 \mathrm{UB}_{\max})$ time (where $\mathrm{UB}_{\max}$ denotes an upper bound on the optimal objective values in both objective functions) and outputs a set of at most $\mathcal{O}(n^2 \cdot \frac{1}{\epsilon^2} \cdot \log^2 \mathrm{UB}_{\max})$ points from its dynamic programming recursion. After all weakly dominated solutions have been removed in the final step, a Pareto curve containing only $\mathcal{O}(n \cdot \frac{1}{\epsilon} \cdot \log \mathrm{UB}_{\max})$ points is produced. Hence, the running time of the procedure used to obtain the parametric solution from the approximate Pareto curve in Proposition 4 is dominated by the running time of the bicriteria FPTAS and we obtain a total running time of $\mathcal{O}(n^3 \cdot \frac{1}{\epsilon^2} \cdot \log^2 \mathrm{UB}_{\max})$ for the resulting parametric FPTAS.

References

[1] H. Kellerer, U. Pferschy, D. Pisinger, Knapsack Problems, Springer, 2004.

[2] P.J. Carstensen, Complexity of some parametric integer and network programming problems, Math. Program. 26 (1) (1983) 64–75.

[3] R.M. Karp, J.B. Orlin, Parametric shortest path algorithms with an application to cyclic staffing, Discrete Appl. Math. 3 (1) (1981) 37–45.

[4] N.E. Young, R.E. Tarjan, J.B. Orlin, Faster parametric shortest path and minimum-balance algorithms, Networks 21 (2) (2006) 205–221.

[5] P.J. Carstensen, The Complexity of Some Problems in Parametric, Linear, and Combinatorial Programming, Ph.D. thesis, University of Michigan, 1983.

[6] K. Mulmuley, P. Shah, A lower bound for the shortest path problem, J. Comput. Syst. Sci. 63 (2) (2001) 253–267.

[7] D. Fernández-Baca, G. Slutzki, D. Eppstein, Using sparsification for parametric minimum spanning tree problems, in: Proceedings of the 5th Scandinavian Workshop on Algorithm Theory, SWAT, in: Lect. Notes Comput. Sci., vol. 1097, 1996, pp. 149–160.

[8] G. Gallo, M.D. Grigoriadis, R.E. Tarjan, A fast parametric maximum flow algorithm and applications, SIAM J. Comput. 18 (1) (1989) 30–55.

[9] T. Arai, S. Ueno, Y. Kajitani, Generalization of a theorem on the parametric maximum flow problem, Discrete Appl. Math. 41 (1) (1993) 69–74.

[10] S.T. McCormick, Fast algorithms for parametric scheduling come from extensions to parametric maximum flow, Oper. Res. 47 (5) (1999) 744–756.

[11] M.G. Scutellà, A note on the parametric maximum flow problem and some related reoptimization issues, Ann. Oper. Res. 150 (1) (2007) 231–244.

[12] A. Mitsos, P.I. Barton, Parametric mixed-integer 0–1 linear programming: the general case for a single parameter, Eur. J. Oper. Res. 194 (3) (2009) 663–686.

[13] M. Eben-Chaime, Parametric solution for linear bicriteria knapsack models, Manag. Sci. 42 (11) (1996) 1565–1575.

[14] M.J. Eisner, D.G. Severance, Mathematical techniques for efficient record segmentation in large shared databases, J. ACM 23 (4) (1976) 619–635.

[15] R.E. Burkhard, U. Pferschy, The inverse-parametric knapsack problem, Eur. J. Oper. Res. 83 (2) (1995) 376–393.

[16] H.M. Safer, J.B. Orlin, Fast Approximation Schemes for Multi-Criteria Combinatorial Optimization, Working Paper 3756-95, Sloan School of Management, Massachusetts Institute of Technology, 1995.

[17] H.M. Safer, J.B. Orlin, Fast Approximation Schemes for Multi-Criteria Flow, Knapsack, and Scheduling Problems, Working Paper 3757-95, Sloan School of Management, Massachusetts Institute of Technology, 1995.

[18] T. Erlebach, H. Kellerer, U. Pferschy, Approximating multiobjective knapsack problems, Manag. Sci. 48 (12) (2002) 1603–1612.

[19] C. Bazgan, H. Hugot, D. Vanderpooten, Implementing an efficient FPTAS for the 0–1 multi-objective knapsack problem, Eur. J. Oper. Res. 198 (1) (2009) 47–56.

[20] S. Sahni, Approximate algorithms for the 0/1 knapsack problem, J. ACM 22 (1) (1975) 115–124.

[21] I.J. Balaban, An optimal algorithm for finding segment intersections, in: Proceedings of the 11th Annual Symposium on Computational Geometry, SOCG, 1995, pp. 211–219.

[22] M. de Berg, O. Cheong, M. van Kreveld, M. Overmars, Computational Geometry: Algorithms and Applications, Springer, 2000.

[23] B. Chazelle, Reporting and counting segment intersections, J. Comput. Syst. Sci. 32 (2) (1986) 156–182.

[24] F.P. Preparata, M. Shamos, Computational Geometry: An Introduction, Springer Science & Business Media, 2012.