ARTICLE IN PRESS

Discrete Applied Mathematics xxx (xxxx) xxx

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

A local analysis to determine all optimal solutions of p-k-max location problems on networks

Teresa Schnepper [a], Kathrin Klamroth [a], Justo Puerto [b], Michael Stiglmayr [a,*]

[a] School of Mathematics and Natural Sciences, University of Wuppertal, Gaußstraße 20, 42103 Wuppertal, Germany

[b] Mathematical Research Institute (IMUS), University of Seville, C/ Tarfia 21, 41012 Seville, Spain

ARTICLE INFO

Article history:
Received 28 May 2019
Received in revised form 9 April 2020
Accepted 12 May 2020
Available online xxxx

Keywords:
Center problem on networks
Outlier
k-max function
Local analysis
Complete optimal set
Secondary objective

ABSTRACT

As a generalization of p-center location problems, p-k-max problems minimize the kth largest weighted distance to the customers. In this way, outlier facilities can be detected automatically and excluded from consideration when locating new facilities. Similar to p-center problems, p-k-max problems often have many alternative optimal solutions. Knowledge of the complete optimal set allows to select a most preferred solution using secondary criteria. In this paper, a general solution method is suggested that guarantees to find all alternative optimal solutions of p-k-max problems on networks. This is realized by performing a local analysis on the edges of the underlying graph and identifying edge segments on which the p-k-max function is linear. It is shown that the complete optimal set can be represented by an extended finite dominating set (FDS) which is of polynomial size for fixed values of p. Numerical tests indicate that computing the set of optimal solutions compared to the computation of a single optimal solution of a p-k-max problem requires on average less than 15% additional computing effort. This computational efficiency allows one to select the most preferred solution among them using secondary objectives, like backup coverage or the Weber function.

1. Introduction

Bottleneck objective functions are in general very sensitive to outliers and data errors. In the context of location problems a small number of far-away customers, which we denote as *outliers*, can influence the solution of location problems such that the distance to the majority of customers is increased dramatically. In economically motivated situations this might lead to unfavorable solutions. Outliers in location problems are mainly not caused by data errors (in contrast to applications in data science) but by very heterogeneous distributions of costumers.

There has been extensive work on different techniques for handling outliers in location problems. As the influence of far-away facilities is most significant for center location problems, most approaches focus on this case. Continuous location problems with outliers are considered, e.g., in [1] for Euclidean distances, and in [2] for l_∞-distances. In [4,5] so-called minquantile location problems are investigated, which can be considered a general case of k-max location problems. More general cases are considered in [30] (high dimensional problems) and in [6] (finite metric spaces). Discrete location models with outliers are discussed, e.g., in [7,28] and in [6].

* Corresponding author.
E-mail addresses: schnepper@math.uni-wuppertal.de (T. Schnepper), klamroth@math.uni-wuppertal.de (K. Klamroth), puerto@us.es (J. Puerto), stiglmayr@math.uni-wuppertal.de (M. Stiglmayr).

https://doi.org/10.1016/j.dam.2020.05.013
0166-218X/

Generalizing the center objective by using the kth largest weighted distance between any customer and its closest new facility instead of the largest weighted distance (where $(k-1) \in \{0, 1, \ldots, n-1\}$ specifies an acceptable number of outliers) naturally limits the influence of far-away facilities. It can thus be applied as a general technique for handling outliers. For a detailed introduction to k-max location problems, see [24] and [23]. The concept of k-max optimization also occurs in the context of combinatorial optimization, see, e.g., [11,22] and [27]. Moreover, k-max functions are a special case of the ordered median objective, see, e.g., [21] and [16].

It is well known that optimization problems with bottleneck objective tend to have many alternative optimal solutions. This can be explained by the fact that bottleneck type objective functions (including k-max functions) usually rely, when evaluated in an optimal solution, on only a small subset of customers/active constraints while all other customers/constraints do not contribute/are inactive. Two ways to handle this ambiguity are the following: Either rank the optimal solutions using, e.g., a lexicographic bottleneck objective function (see [3,25]) or compute all alternative optimal solutions (see, e.g., [26,29]) and then apply a subsequent decision making process. A prominent example in the context of location analysis is the maximization of backup coverage, see, e.g., [13].

In this paper, a general approach for the determination of the set of *all* optimal solutions of p-k-max problems on networks is developed. Based on a detailed geometric analysis of the k-max function, including the identification of linearity regions, a finite dominating set (FDS) is identified that can be used to represent the optimal set. It is shown that the FDS has polynomial size as long as the number p of new facilities is fixed, and that it is usually relatively small in practice. Moreover, a reduced FDS can be used to define seed points for a local analysis, also leading to a complete description of the optimal set. It turns out that the local analysis is independent of the parameter k, and thus the optimal sets for different values of k can be determined in parallel. Computational tests indicate that the potential gain w.r.t. secondary objectives as, for example, total transportation cost, double coverage and evenly distributed capacities is significant when a most preferred solution can be selected from the complete optimal set.

The reminder of this paper is organized as follows: Section 2 summarizes the notation and provides a formal definition of k-max problems on graphs. In Section 3, an example illustrates the situation and some known results for p-k-max problems are reviewed. Based on a local analysis, a finite dominating set leading to all optimal solutions of the problem is deduced and further reduced in Section 4. The evaluation of these candidate sets is extensively tested and the results are discussed in Section 5. The paper concludes with a short summary in Section 6.

2. Notation and problem definition

The p-k-max problem as considered in this paper is defined on a simple, connected and undirected graph $G = (V_G, E)$ with node set $V_G = \{v_1, \ldots, v_{n_G}\}$ and edge set $E = \{e_1, \ldots, e_m\} \subseteq V_G \times V_G$. Edges can be represented by their index, i.e., $e_j \in E$, or equivalently by their end nodes, i.e., $e_j = e_{ab} = [v_a, v_b]$. The set of customers, also referred to as clients, is given by a subset $V = \{v_1, \ldots, v_n\} \subseteq V_G$, with $n \le n_G$. The demand of customer $v_i \in V,\ i = 1, \ldots, n$ is represented by a strictly positive weight $w_i > 0$. Distances are modeled by assigning a strictly positive length $l_j = l_{ab} > 0$ to every edge $e_j = e_{ab} \in E$, and are assumed to be linear along the edges.

New facilities may be located in nodes and on edges. The continuum set of points on the edges of G is denoted by $A(G)$, with $V_G \subseteq A(G)$. A point x on an edge $e_{ab} \in E$ is represented by a pair $x = (e_{ab}, t)$ with $t \in [0, 1]$, and it is equivalently identified by the parameter $t \in [0, 1]$ as long as the edge e_{ab} is fixed. Let the integer $p \in \{1, \ldots, n\}$ specify the number of new facilities to be located. Then $X = \{x_1, \ldots, x_p\} \subseteq A(G)$ denotes a feasible set of p new facilities $x_1, \ldots, x_p \in A(G)$.

Now consider an edge $e_{ab} \in E$. The distance between a point $x = (e_{ab}, t)$ on e_{ab} and an arbitrary customer $v_i \in V$ is given by

$$d(v_i, x) = \min\left\{d(v_i, v_a) + t\, l_{ab},\, d(v_i, v_b) + (1-t) l_{ab}\right\},$$

where $d(v_i, v_a)$ denotes the length of the shortest path between v_i and v_a. Note that $d(v_i, x)$ defines a metric on G. Accordingly, the distance from a customer $v_i \in V$ to a set of new facilities X can be computed as

$$d(v_i, X) = \min_{x \in X} d(v_i, x).$$

The corresponding weighted distances from a customer $v_i \in V$ are denoted by $d^w(v_i, x) = w_i\, d(v_i, x)$ and $d^w(v_i, X) = w_i\, d(v_i, X)$, respectively. The component-wise weighted distances between the finite set of customers V and the respective closest new facility in the set X is then given by the vector

$$d^w(V, X) = \left(d^w(v_1, X), \ldots, d^w(v_n, X)\right)^\top.$$

Let $k \in \{1, \ldots, n\}$ denote the parameter of the k-max function that specifies the kth largest weighted distance to be minimized. Then the p-k-max problem models the situation of locating a set $X = \{x_1, \ldots, x_p\} \subseteq A(G)$ of p new facilities such that the kth largest weighted distance between the set of customers V and the set of new facilities X is minimized. For a fixed set X of new facilities, the kth largest weighted distance between X and V can be identified using a permutation σ of the customers that satisfies

$$d^w(v_{\sigma(1)}, X) \ge d^w(v_{\sigma(2)}, X) \ge \cdots \ge d^w(v_{\sigma(n)}, X). \tag{1}$$

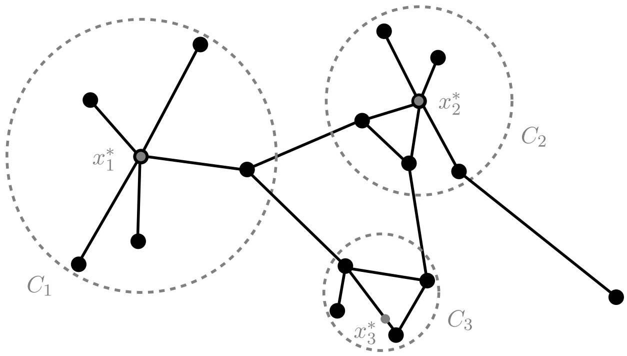


Fig. 1. An instance of a 3-2-max problem with an optimal solution $X^* = \{x_1^*, x_2^*, x_3^*\}$.

Note that the permutation σ is in general not unique and that it depends on the current solution X. The set $\Sigma(X)$ of all permutations satisfying (1) is referred to as the set of *valid permutations* w.r.t. X. The *p-k*-max *problem* is now given by

$$\min_{\substack{X\subseteq A(G)\\|X|=p}} k\text{-max}(d^w(V,X)) = \min_{\substack{X\subseteq A(G)\\|X|=p}} w_{\sigma(k)}\, d(v_{\sigma(k)}, X) \quad \text{for } \sigma \in \Sigma(X). \tag{pkMG}$$

In [24] it is shown that the p-k-max problem always has an optimal solution. Since the set of optimal solutions for p-k-max problems with $k \geq n-p+1$ can be given explicitly [24], only p-k-max problems with $k \leq n-p$ are considered in the following.

3. Finite dominating sets: Finding specific optimal solutions

The p-k-max problem (pkMG) can be interpreted as a p-center problem with the automatic detection of $(k-1)$ outliers. Outlier sets thus play a decisive role in this context. For a fixed solution $X = \{x_1, \dots, x_p\}$ and a valid permutation $\sigma \in \Sigma(X)$, the set

$$V_{k-1} = \{v_{\sigma(1)}, \dots, v_{\sigma(k-1)}\} \subseteq V$$

is referred to as an *outlier set* associated to X. If V_{k-1} is not unique, it is assumed that V_{k-1} is chosen arbitrarily but fixed. The customers in the set $V \setminus V_{k-1}$ are called *center defining customers* w.r.t. X. An outlier set associated to an optimal solution X^* is denoted by V_{k-1}^* and called an *optimal outlier set*. Moreover, each new facility $x_i \in X$, $i = 1, \dots, p$, gives rise to a set $C_i \subseteq V \setminus V_{k-1}$ of customers that are covered by the respective new facility. To avoid ambiguities, each center defining customer is allocated to the lowest numbered facility that covers it, i.e., the *clusters* C_i are defined as:

$$\begin{aligned} C_i = \Big\{ v \in V \setminus V_{k-1} : d^w(v, x_i) \leq d^w(v, x_\ell)\ \forall\, \ell > i \\ \wedge\ d^w(v, x_i) < d^w(v, x_\ell)\ \forall\, \ell < i \Big\}, \quad i = 1, \dots, p. \end{aligned} \tag{2}$$

The maximum distance between a new facility $x_i \in X$, $i \in \{1, \dots, p\}$ and a most distant customer in the associated cluster C_i is called the *radius* of the cluster C_i, i.e., $r_i = \max_{v \in C_i} d^w(v, x_i)$.

Example 3.1. An instance of a 3-2-max problem is illustrated in Fig. 1. The set of customers is given by $V = V_G$ and all customers are weighted equally, i.e., $w_i = 1$ for all $v_i \in V$. Edge lengths l_j, $e_j \in E$, correspond to the Euclidean distances in the depicted embedding of G.

Fig. 1 shows an optimal solution $X^* = \{x_1^*, x_2^*, x_3^*\}$ together with the respective clusters C_1, C_2, C_3. An optimal outlier set is given by the node that is not contained in any of the three clusters, i.e., $V_1^* = V \setminus (C_1 \cup C_2 \cup C_3)$. Accordingly, the set of center defining customers w.r.t. X^* is given by $V \setminus V_1^* = C_1 \cup C_2 \cup C_3$.

Example 3.1 illustrates that the k-max value z of the solution X^* is defined by a cluster with largest radius, in this case by the cluster C_1. In the following, a new facility $x_i \in X$, $i \in \{1, \dots, p\}$ for which $r_i = z$ is called an *objective value defining facility*. In Example 3.1, the objective value defining facility x_1^* is unique. However, in general an objective value defining facility does not have to be unique, and several clusters may have the same radius.

Similar to the classical p-center problem (see, e.g., [12,18,19]), *finite dominating sets* (FDS) are a powerful tool to efficiently determine at least one optimal solution of the p-k-max problem. Schnepper et al. [24] suggest a polynomial time recursive algorithm for the determination of a subset of optimal solutions that is based on the computation of so-called *equilibrium points*, first defined in [20]. The result is based on the fact that the k-max objective function is piecewise linear on $A(G)^p$ and that equilibrium points can be used to specify candidate locations yielding local minima. In order to compute the complete optimal set, however, the linearity regions of the objective function have to be specified more precisely. Definition 3.2 identifies *equilibrium points* and *bottleneck points*, respectively, as critical point sets in this context. An FDS for minquantile location problems in the plane is suggested in [5] which can be extended to location problems on networks yielding an FDS similar to the one in [24].

Definition 3.2.

(a) For $v_i, v_j \in V$, $i \neq j$, let

$$EQ'_{ij} = \left\{x \in A(G) : w_i\, d(v_i, x) = w_j\, d(v_j, x)\right\},$$

and let EQ_{ij} be the relative boundary of EQ'_{ij}. Then the set of *equilibrium points* of G is given by $EQ := \bigcup_{v_i, v_j \in V,\, i \neq j} EQ_{ij}$.

(b) $x = (e_{ab}, t)$ on $e_{ab} \in E$ is called a *bottleneck point* w.r.t. $v_i \in V$, if

$$w_i(d(v_i, v_a) + d(v_a, x)) = w_i(d(v_i, v_b) + d(v_b, x)) = d^w(v_i, x).$$

The set BN_i denotes the set of all bottleneck points w.r.t. $v_i \in V$ and $BN := \bigcup_{i=1}^n BN_i$ is the set of all bottleneck points of G.

Now the FDS results of Schnepper et al. [24] can be summarized and extended as follows:

Theorem 3.3. *Consider an instance of a p-k-max-problem with $n \geq 2$, $p \in \{1, \ldots, n\}$ and $k \in \{1, \ldots, n-p\}$.*

(a) If $p = 1$, then all optimal solutions can be found in the set EQ.
(b) If $p \geq 2$, then at least one optimal solution $X^* = \{x_1^*, \ldots, x_p^*\}$, where w.l.o.g. x_1^* is an objective value defining facility, can be found in the set $\mathscr{C}^{p\geq 2} := EQ \times (EQ \cup V)^{p-1}$. Moreover, every optimal solution has an objective value defining facility in the set EQ.

Proof. Statement (a) and the first part of statement (b) are proven in [24]. It remains to show that *every* optimal solution has an objective value defining facility in the set EQ.

Towards this end, let $X^* = \{x_1^*, \ldots, x_p^*\} \subseteq A(G)$ be an optimal solution with associated outlier set V_{k-1}^*. Let C_ℓ for all $\ell \in \{1, \ldots, p\}$ denote the clusters of the new facilities $x_1^*, \ldots, x_p^*$ according to (2). Then X^* is an optimal solution of the p-center problem with customers $V \setminus V_{k-1}^*$ (see Theorem 3.1. in [24]). Moreover, at least one objective value defining facility is an optimal 1-center of its respective cluster since otherwise a better solution could easily be constructed by moving these facilities to the respective 1-centers. Since $k \leq n - p$ and $|V \setminus V_{k-1}^*| \geq p + 1$, at least one of the clusters C_ℓ, $\ell \in \{1, \ldots, p\}$ contains at least two customers, and hence this is also the case for the clusters of all objective value defining facilities. That X^* has the desired property now follows from statement (a) of Theorem 3.3 for $k = p = 1$. □

The cardinality of the FDS from Theorem 3.3 can be bounded by $|\mathscr{C}^{p\geq 2}| \in \mathcal{O}(m^p n^{2p})$. Schnepper et al. [24] suggest a recursive algorithm that determines all optimal solutions in the set $\mathscr{C}^{p\geq 2}$ in at most $\mathcal{O}(m^p n^{3p})$ time. Note that the set of optimal solutions $\mathcal{X}_{\mathscr{C}^{p\geq 2}}$ found with this approach does, in general, not contain *all* optimal solutions of the p-k-max problem. However, all objective value defining facilities that lead to at least one optimal solution of the p-k-max problem are known after the evaluation of the set $\mathscr{C}^{p\geq 2}$.

4. Local analysis: Finding all optimal solutions

In this section, the FDS $\mathscr{C}^{p\geq 2}$ from Theorem 3.3 is extended to a new candidate set that is a superset of $\mathscr{C}^{p\geq 2}$ and that contains alternative optimal solutions which have other properties than the candidates in $\mathscr{C}^{p\geq 2}$. The aim of this section is to determine the set $\mathcal{X}^*$ of *all* optimal solutions of the p-k-max problem. The idea is to perform a local analysis over each edge of the graph in order to generate a subdivision of $A(G)$ such that the k-max objective function is piecewise linear and concave on every cell of this subdivision. The resulting FDS can afterwards be reduced by using the objective value defining facilities obtained with the FDS $\mathscr{C}^{p\geq 2}$.

The general approach of a local analysis and the notation used in the following are similar to that of Kalcsics [14], who derived an FDS for the multi-facility median problem with positive and negative weights on general graphs, and introduced an efficient solution procedure based on analyzing regions of fixed allocations with the help of an arrangement of hyperplanes in $\mathbb{R}^p$. A similar approach is also used in [15] for multicriteria p-median problems and in [17] for location problems with equity objectives.

An example with $p = 2$ new facilities is used to illustrate the concept. Note, however, that the approach is applicable in general, i.e., for all $p \geq 2$.

Example 4.1. Consider the weighted graph G shown in Fig. 2. An optimal solution of the 2-1-max problem is

$$X = \left\{\left(e_{34}, \frac{1}{3}\right), \left(e_{12}, \frac{1}{3}\right)\right\} \in \mathscr{C}^{p\geq 2}$$

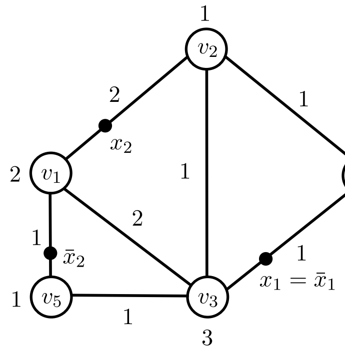


Fig. 2. Two optimal solutions $X \in \mathscr{C}^{p\geq 2}$ and $\bar{X} \notin \mathscr{C}^{p\geq 2}$ of a 2-1-max problem.

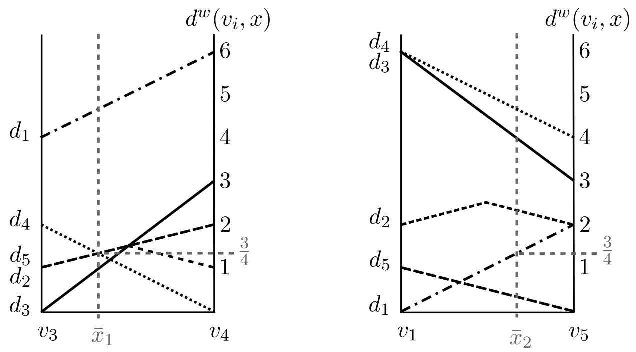


Fig. 3. Distance functions $d^w(v_i, x)$ over the edges e_{34} (left) and e_{15} (right), with $d_i = d^w(v_i, x)$, $i = 1, \ldots, 5$.

with clusters $C_1 = \{v_2, v_3, v_4, v_5\}$ and $C_2 = \{v_1\}$, and with optimal objective value $z = \frac{4}{3}$ (which can be determined by enumerating $\mathscr{C}^{p\geq 2}$). An alternative optimal solution is $\bar{X} = \{\bar{x}_1, \bar{x}_2\}$ given by

$$\bar{x}_1 = \left(e_{34}, \frac{1}{3}\right) \in EQ_{45} \subseteq EQ \quad \text{and} \quad \bar{x}_2 = \left(e_{15}, \frac{2}{3}\right) \notin EQ \cup V$$

associated outlier set is empty. Obviously, $\bar{X}$ cannot be found using the set $\mathscr{C}^{p\geq 2}$ as $\bar{x}_2$ is neither an equilibrium point nor a node of G.

To characterize optimal solutions not lying in the FDS $\mathscr{C}^{p\geq 2}$, weighted distances to customers are analyzed over the edges of G. Given an edge $e_{ab} \in E$, the weighted distance $d^w(v_i, x)$ between a new facility $x = (e_{ab}, t)$ (with $t \in [0, 1]$) and a customer $v_i \in V$ can be interpreted as a function of the parameter t which is piecewise linear and concave with breakpoints only at bottleneck points.

Example 4.2 (*Continuation of Example 4.1*). Fig. 3 shows the graphs of the weighted distance functions over the edges $e_{34} = (v_3, v_4)$ and $e_{15} = (v_1, v_5)$ which contain the two new facilities $\bar{x}_1$ and $\bar{x}_2$, respectively. Note that $\bar{x}_1 \in EQ$ is an objective value defining facility. In this particular solution, $\bar{x}_2$ is also objective value defining, and it holds that

$$d^w(v_2, \bar{x}_1) = d^w(v_4, \bar{x}_1) = d^w(v_1, \bar{x}_2) = \frac{4}{3} \quad \text{and} \quad \bar{r}_1 = \bar{r}_2,$$

i.e., both clusters $\bar{C}_1$ and $\bar{C}_2$ have the same radius. This results in

$$d^w(V, \bar{X}) = \left(\frac{4}{3}, \frac{4}{3}, 1, \frac{4}{3}, \frac{1}{3}\right)^\top,$$

i.e., there are at least three equal elements in the vector of distances. As a consequence, there are several valid permutations w.r.t. $\bar{X}$, for example, $\sigma_1 = (1, 2, 4, 3, 5)$ and $\sigma_2 = (1, 4, 2, 3, 5)$. With just a small shift of one of the new facilities (for example, moving $\bar{x}_2$ towards v_5), this equality of weighted distances is broken and at least one of the permutations becomes invalid. This induces a non-linearity of the k-max objective function in $\bar{X}$.

4.1. A finite dominating set based on linearity regions

In the following, let $X = \{x_1, \ldots, x_p\}$ be an arbitrary but fixed set of p new facilities with $x_1, \ldots, x_p \in A(G)$. Moreover, let $x_q = (e_{g_q}, t_q)$ with $e_{g_q} = (v_{a_{g_q}}, v_{b_{g_q}})$ for all $q = 1, \ldots, p$ and $g_q \in \{1, \ldots, m\}$. For the weighted distance between a customer $v_i \in V$ and X it holds that

$$\begin{aligned} d^w(v_i, X) = \min &\left\{d_+^w(v_i, x_1), d_-^w(v_i, x_1), \ldots, d_+^w(v_i, x_p), d_-^w(v_i, x_p)\right\}, \\ &\text{where } d_+^w(v_i, x_j) = w_i\,(d(v_i, v_{a_{g_j}}) + t_j\, l_{g_j}) \\ &\text{and } d_-^w(v_i, x_j) = w_i\,(d(v_i, v_{b_{g_j}}) + (1-t_j)\, l_{g_j}) \quad \text{for } j = 1, \ldots, p \end{aligned}$$

The valid ordering of the components of the vector of weighted distances $d^w(V, X)$ is unique when none of its components are equal, and it may change whenever at least two of its components are equal, i.e., whenever

$$d_\alpha^w(v_i, x_q) = d_\beta^w(v_j, x_r) \tag{3}$$

for some $i, j \in \{1, \ldots, n\}$, $q, r \in \{1, \ldots, p\}$, $\alpha, \beta \in \{+, -\}$ (where at least one of the pairs i, j, q, r and α, β is not identical). Similarly, an individual weighted distance $d^w(v_i, X)$ is piecewise linear with possible breakpoints only when (3) is satisfied (with $j = i$). Hence, all points satisfying (3) may correspond to breakpoints of the objective function of the p-k-max problem.

The above analysis suggests that the linearity regions, or rather their boundaries, induce an FDS for the p-k-max problem which is a superset of $\mathscr{C}^{p\geq 2}$ and which potentially contains further optimal solutions. This FDS can be described based on a subdivision of the unit hypercube $[0, 1]^p$. More precisely, p-tuples of edges $e_{g_1}, \ldots, e_{g_p} \in E$ with $g_1 \leq g_2 \leq \cdots \leq g_p$ are considered, where each edge is identified with the unit interval $[0, 1]$. The Cartesian product $e_{g_1} \times \cdots \times e_{g_p}$ is represented by the unit hypercube $[0, 1]^p$. Let $g = (g_1, \ldots, g_p) \in \{1, \ldots, m\}^p$ indicate an arbitrary but fixed p-tuple of edges. The p new facilities $x_1 = (e_{g_1}, t_1), \ldots, x_p = (e_{g_p}, t_p)$ are assumed to be located on the edges $e_{g_1}, \ldots, e_{g_p}$, i.e., $X \subseteq A(G)$ for short (by slightly abusing the notation). Note that since the ordering of the new facilities in $X = \{x_1, \ldots, x_p\}$ is not relevant, all feasible solutions can be associated with a p-tuple of edges $g = (g_1, \ldots, g_p)$ and with a point $(t_1, \ldots, t_p) \in [0, 1]^p$ in the corresponding unit hypercube. However, since several components of g (and thus edges in the corresponding p-tuple) may be equal, this is in general not a one-to-one correspondence, i.e., one solution X may correspond to several points in the associated unit hypercube if several new facilities are located on the same edge.

Now the linearity regions of the k-max function can be described by a subdivision of all relevant p-tuples g of edges and their associated unit hypercubes $[0, 1]^p$. Let g be arbitrary but fixed. Then all breakpoints of the k-max function on $[0, 1]^p$ must satisfy Eq. (3). In the following, *all* solutions of Eq. (3) in $[0, 1]^p$ will be determined by distinguishing four possible cases for the choice of the indices $i, j \in \{1, \ldots, n\}$ and $q, r \in \{1, \ldots, p\}$. Note that the point set determined in this way certainly contains all boundaries of linearity regions of the k-max function.

Case 1: $i \neq j$ and $q = r$, i.e., $d_\alpha^w(v_i, x_q) = d_\beta^w(v_j, x_q)$ for $i, j \in \{1, \ldots, n\}$, $i \neq j$, $q \in \{1, \ldots, p\}$ and $\alpha, \beta \in \{+, -\}$. In this case, the set of breakpoints is given by

$$B_{ijqq}^{\alpha,\beta} := \left\{\{x_1, \ldots, x_p\} \subset (e_{g_1} \cup \cdots \cup e_{g_p}) : x_q = EQ_{ij}^{e_{g_q}}(\alpha, \beta)\right\},$$

where $EQ_{ij}^{e_{g_q}}(\alpha, \beta)$ denotes the unique equilibrium point on $e_{g_q} = (v_{a_{g_q}}, v_{b_{g_q}})$ for v_i, v_j and for a fixed combination of $\alpha, \beta \in \{+, -\}$ (if it exists on e_{g_q}). Analyzing all four possible combinations of α and β leads to four equations describing the corresponding locations of $x_q = (e_{g_q}, t_q)$:

$$t_q = \frac{w_j d(v_j, v_{a_{g_q}}) - w_i d(v_i, v_{a_{g_q}})}{w_i l_{g_q} - w_j l_{g_q}} =: t_q^{++}, \quad \text{if } w_i \neq w_j \tag{4}$$

$$t_q = \frac{w_j d(v_j, v_{b_{g_q}}) + w_j l_{g_q} - w_i d(v_i, v_{a_{g_q}})}{w_i l_{g_q} + w_j l_{g_q}} =: t_q^{+-} \tag{5}$$

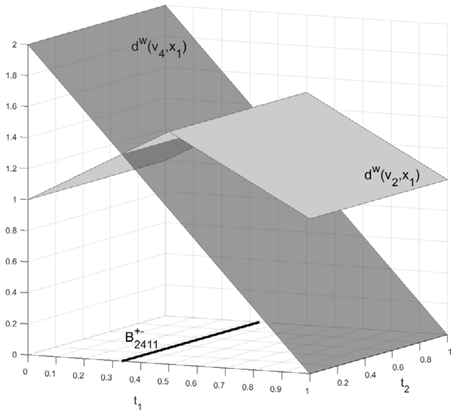
$$t_q = \frac{w_j d(v_j, v_{a_{g_q}}) - w_i d(v_i, v_{b_{g_q}}) - w_i l_{g_q}}{-w_i l_{g_q} - w_j l_{g_q}} =: t_q^{-+} \tag{6}$$

$$t_q = \frac{w_j d(v_j, v_{b_{g_q}}) + w_j l_{g_q} - w_i d(v_i, v_{b_{g_q}}) - w_i l_{g_q}}{-w_i l_{g_q} + w_j l_{g_q}} =: t_q^{--}, \quad \text{if } w_i \neq w_j. \tag{7}$$

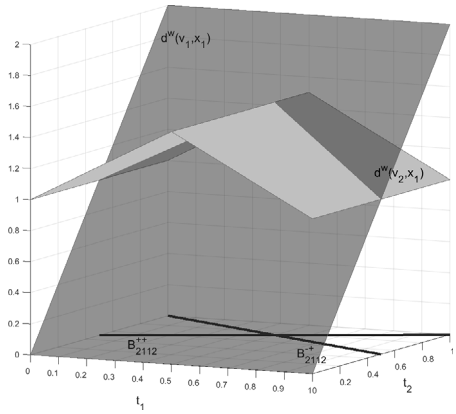
Note that the set $B_{ijqq}^{\alpha,\beta}$ corresponds to the intersection of $[0, 1]^p$ with a hyperplane which is given by $\{(t_1, \ldots, t_p) \in \mathbb{R}^p : t_q = t_q^{\alpha,\beta}\}$, $\alpha, \beta \in \{+, -\}$. Such hyperplanes are called *equilibrium hyperplanes* in the following, see Fig. 4a for an illustration.

Case 2: $i \neq j$ and $q \neq r$, i.e., $d_\alpha^w(v_i, x_q) = d_\beta^w(v_j, x_r)$ for $i, j \in \{1, \ldots, n\}$, $i \neq j$, $q, r \in \{1, \ldots, p\}$, $q \neq r$, and $\alpha, \beta \in \{+, -\}$. The set of breakpoints in this case is given by

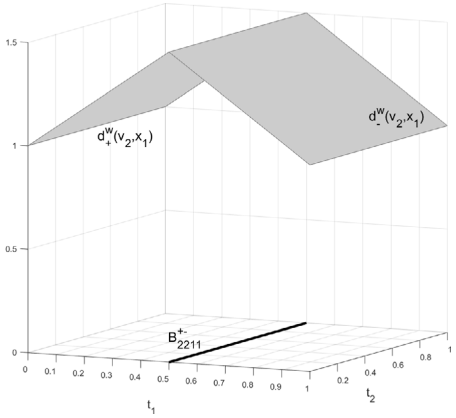
$$B_{ijqr}^{\alpha\beta} := \left\{\{x_1, \ldots, x_p\} \subset (e_{g_1} \cup \cdots \cup e_{g_p}) : d_\alpha^w(v_i, x_q) = d_\beta^w(v_j, x_r)\right\},$$



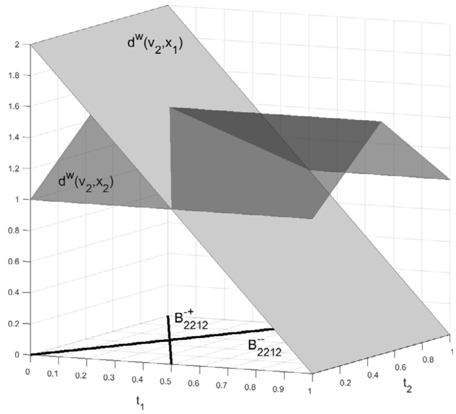
(a) Equilibrium hyperplane (line) of B^{+-}_{2411} for $e_{34} \times e_{15}$, i.e., $x_1 = (e_{34}, t_1)$, $x_2 = (e_{15}, t_2)$



(b) Balance hyperplanes (lines) given by B^{++}_{2112} and B^{-+}_{2112} for $e_{34} \times e_{15}$, i.e., $x_1 = (e_{34}, t_1)$, $x_2 = (e_{15}, t_2)$



(c) Bottleneck hyperplane (line) of B^{+-}_{2211} for $e_{34} \times e_{15}$, i.e., $x_1 = (e_{34}, t_1)$, $x_2 = (e_{15}, t_2)$



(d) Balance hyperplanes (lines) of B^{-+}_{2212} and B^{--}_{2212} for $e_{12} \times e_{34}$, i.e., $x_1 = (e_{12}, t_1)$, $x_2 = (e_{34}, t_2)$

Fig. 4. Some boundaries of linearity regions for the instance introduced in Example 4.1.

for $\alpha, \beta \in \{+, -\}$. The set $B^{\alpha\beta}_{ijqr}$ contains all solutions that have two facilities on the edges e_{g_q} and e_{g_r}, respectively, which have the same weighted distance to two different customers v_i and v_j. Analyzing again all four cases of α-β-combinations leads to the following equations in $t_q, t_r \in [0, 1]$ that describe the corresponding locations of $x_q = (e_{g_q}, t_q)$ and $x_r = (e_{g_r}, t_r)$:

$$w_i\, l_{g_q}\, t_q - w_j\, l_{g_r}\, t_r = w_j\, d(v_j, v_{a_{g_r}}) - w_i\, d(v_i, v_{a_{g_q}}) \tag{8}$$

$$w_i\, l_{g_q}\, t_q + w_j l_{g_r} t_r = w_j\, d(v_j, v_{b_{g_r}}) - w_i\, d(v_i, v_{a_{g_q}}) + w_j\, l_{g_r} \tag{9}$$

$$-w_i\, l_{g_q}\, t_q - w_j l_{g_r} t_r = w_j\, d(v_j, v_{a_{g_r}}) - w_i\, d(v_i, v_{b_{g_q}}) - w_i\, l_{g_q} \tag{10}$$

$$-w_i\, l_{g_q}\, t_q + w_j\, l_{g_r}\, t_r = w_j\, d(v_j, v_{b_{g_r}}) - w_i\, d(v_i, v_{b_{g_q}}) - w_i\, l_{g_q} + w_j\, l_{g_r}. \tag{11}$$

The hyperplanes in $\mathbb{R}^p$ defined by these equations (c.f. Case 1 above) are called *balance hyperplanes*, see Fig. 4b for an illustration.

Case 3: $i = j$ and $q = r$, i.e., $d^w_\alpha(v_i, x_q) = d^w_\beta(v_i, x_q)$ such that $i \in \{1, \dots, n\}$, $q \in \{1, \dots, p\}$ and $\alpha, \beta \in \{+, -\}$, $\alpha \neq \beta$. W.l.o.g. let $(\alpha, \beta) = (+, -)$. The set of breakpoints is then given by

$$B^{+-}_{iiqq} := \left\{ \{x_1, \dots, x_p\} \subset (e_{g_1} \cup \dots \cup e_{g_p}) : x_q = BN_i^{e_{g_q}} \right\},$$

where $BN_i^{e_{g_q}}$ denotes the unique bottleneck point on $e_{g_q} = (v_{a_{g_q}}, v_{b_{g_q}})$ for v_i (if it exists on e_{g_q}). Solving Eq. (3) for t_q to obtain the corresponding location of $x_q = (e_{g_q}, t_q)$ leads to

$$t_q = \frac{w_i d(v_i, v_{b_{g_q}}) + w_i l_{g_q} - w_i d(v_i, v_{a_{g_q}})}{2 w_i l_{g_q}}. \tag{12}$$

The hyperplanes in $\mathbb{R}^p$ defined by these equations are called *bottleneck hyperplanes*, see Fig. 4c for an illustration.

Case 4: $i = j$ and $q \neq r$, i.e., $d_\alpha^w(v_i, x_q) = d_\beta^w(v_i, x_r)$ for $i \in \{1, \dots, n\}$, $q, r \in \{1, \dots, p\}$, $q \neq r$, and $\alpha, \beta \in \{+, -\}$. Note that this case can be integrated in Case 2 above. However, since it corresponds to a different geometric situation, we discuss it here for the sake of clarity. The set of breakpoints in this case is described by the set

$$B_{iiqr}^{\alpha\beta} := \left\{ \{x_1, \dots, x_p\} \subset (e_{g_1} \cup \dots \cup e_{g_p}) : d_\alpha^w(v_i, x_q) = d_\beta^w(v_i, x_r) \right\}.$$

This set contains pairs of new facilities $x_q = (e_{g_q}, t_q)$ and $x_r = (e_{g_r}, t_r)$ that have the same distance to the customer v_i. The corresponding equations in $t_q, t_r \in [0, 1]$, derived from all possible α-β-combinations, are:

$$l_{g_q} t_q - l_{g_r} t_r = d(v_i, v_{a_{g_r}}) - d(v_i, v_{a_{g_q}}) \tag{13}$$

$$l_{g_q} t_q + l_{g_r} t_r = d(v_i, v_{b_{g_r}}) - d(v_i, v_{a_{g_q}}) + l_{g_r} \tag{14}$$

$$-l_{g_q} t_q - l_{g_r} t_r = d(v_i, v_{a_{g_r}}) - d(v_i, v_{b_{g_q}}) - l_{g_q} \tag{15}$$

$$-l_{g_q} t_q + l_{g_r} t_r = d(v_i, v_{b_{g_r}}) - d(v_i, v_{b_{g_q}}) - l_{g_q} + l_{g_r}. \tag{16}$$

Due to the similarity to Case 2, the corresponding hyperplanes in $\mathbb{R}^p$ are also called *balance hyperplanes*, see Fig. 4d for an illustration.

Note that Eqs. (4)–(16) describe hyperplanes in $\mathbb{R}^p$. Only the intersection of these hyperplanes with the unit hypercube (which may be empty) are of interest since $t_1, \dots, t_p$ have to be in $[0, 1]$ to describe a point on an edge of the graph. In Cases 1 and 3 these hyperplanes are parallel to a face of $[0, 1]^p$.

In the following, let again $g = (g_1, \dots, g_p)$ be an arbitrary but fixed vector of indices of edges $e_{g_1}, \dots, e_{g_q}$ with $g_1 \leq g_2 \leq \dots \leq g_p$. The set of equilibrium, bottleneck and balance hyperplanes for a fixed pair of customers $i, j \in \{1, \dots, n\}$ is denoted by L_{ijg}. By $\bigcup_{i,j \in \{1,\dots,n\}} L_{ijg}$ the set of all hyperplanes for all pairs of customers is described:

$$L_g = \bigcup_{i,j \in \{1,\dots,n\}} L_{ijg} \cup \bigcup_{\ell \in \{1,\dots,p\}} \{t \in \mathbb{R}^p : t_\ell = 0\} \cup \bigcup_{\ell \in \{1,\dots,p\}} \{t \in \mathbb{R}^p : t_\ell = 1\}$$

The arrangement of hyperplanes L_g defines a subdivision of the unit cube $[0, 1]^p$. Any non-empty intersection of the unit cube $[0, 1]^p$ with a set of halfspaces corresponding to the hyperplanes in L_g is a face of the subdivision. A face is denoted as j-face if its dimension is j. In particular, a p-face is called a *cell* $C \subset [0, 1]^p$ of the arrangement and the set $C(L_g)$ denotes the set of all cells of L_g. The extreme points of a cell $C \in C(L_g)$ are 0-faces, which are also called *vertices*. The set of all vertices of L_g is denoted by $V(L_g)$. The $(p-1)$-faces are also called *facets* of the arrangement. Note that $V(L_g)$ coincides with the set of all 0-dimensional intersections of hyperplanes in L_g. For an illustration see Fig. 5, and for more information on arrangements of hyperplanes see, for example, [8].

Let L'_g be the arrangement of hyperplanes in $[0, 1]^p$ that is defined only by the boundary faces of $[0, 1]^p$ and the equilibrium and the balance hyperplanes (Cases 1, 2 and 4), see Fig. 6.

Theorem 4.3. *Let $g = (g_1, \dots, g_p)$ be an arbitrary but fixed vector of indices of edges $e_{g_1}, \dots, e_{g_p}$ with $g_1 \leq g_2 \leq \dots \leq g_p$. Then the objective function of the p-k-max problem on graphs is linear on every cell $C \in C(L_g)$ and it is concave on every cell $C' \in C(L'_g)$.*

Proof. Follows from the construction of the equilibrium, bottleneck, and balance hyperplanes: Consider an arbitrary solution $X \subseteq A(G)$. Then the valid permutation $\sigma \in \Sigma(X)$ is unique (and remains unchanged in a neighborhood of X in $A(G)$) as long as X is not on any equilibrium hyperplane according to Case 1, and not on any balance hyperplane according to Case 2. Similarly, the weighted distances $d^w(v_i, X)$, $i \in \{1, \dots, n\}$ are piecewise linear over $[0, 1]^p$ and may have breakpoints only on bottleneck hyperplanes according to Case 3, and on balance hyperplanes according to Case 4. Thus, the k-max function is linear on every cell $C \in C(L_g)$. Furthermore, bottleneck hyperplanes describe maxima of the weighted distance functions the k-max function is concave on every cell $C' \in C(L'_g)$. □

Theorem 4.4. *Let $n \geq 2$, $p \in \{1, \dots, n\}$ and $k \in \{1, \dots, n-p\}$. At least one optimal solution of the p-k-max problem can be found in the set*

$$V(L') = \bigcup_{\substack{g \in \{1,\dots,m\}^p, \\ g_1 \leq \dots \leq g_p}} V(L'_g).$$

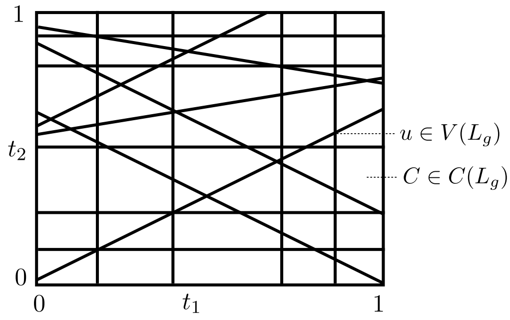


Fig. 5. Illustration of an exemplary subdivision L_g over $[0, 1]^2$ induced by equilibrium, bottleneck and balance lines.

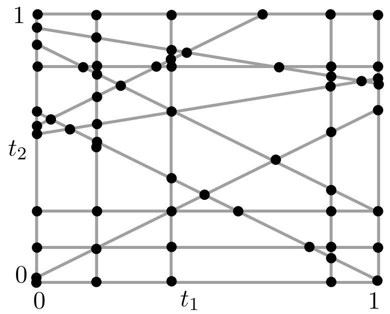


Fig. 6. Illustration of an exemplary subdivision L'_g over $[0, 1]^2$ deduced from L_g by eliminating all bottleneck lines. The points indicate the set $V(L')$.

Proof. Let $g = (g_1, \ldots, g_p)^T \in \{1, \ldots, m\}^p$ with $g_1 \leq \cdots \leq g_p$ be arbitrary but fixed. Following Theorem 4.3, the k-max function is concave on each cell C' of the subdivision L'_g of the unit hypercube $[0, 1]^p$. Moreover, every cell is convex and compact. Thus, a local minimum of the k-max function is attained on the boundary of a cell $C' \in C(L'_g)$. In particular, the set $V(L'_g)$ contains a local optimal solution of the p-k-max-problem restricted to the current p-tuple g. $\square$

Remark 4.5. Note that since the k-max function is piecewise linear and concave on each cell $C' \in C(L'_g)$, $V(L')$ forms a *skeleton* of optimal solutions, i.e. it contains at least one solution in each connected component of $\mathcal{X}^*$. However, we use in the following a refined subdivision into linearity regions to determine the complete optimal set. This is the topic of Section 4.3.

Note that the set $V(L) = \bigcup_{g \in \{1,\ldots,m\}^p, g_1 \leq \cdots \leq g_p} V(L_g)$ is of course also an FDS for the p-k-max problem as it is a superset of $V(L')$.

For a fixed vector $g \in \{1, \ldots, m\}^p$, $g_1 \leq \cdots \leq g_p$, of edge indices, there exist at most $\mathcal{O}(n^2 p)$ equilibrium hyperplanes, $\mathcal{O}(n\,p)$ bottleneck hyperplanes and $\mathcal{O}(n^2 p^2)$ balance hyperplanes. This adds up to at most $\mathcal{O}(n^2 p^2)$ hyperplanes that constitute L_g. To determine the number of candidates resulting from the subdivision L_g, the number of 0-faces of the arrangement is needed. Following [8], the number of 0-faces of an arrangement of $\mathcal{O}(n^2 p^2)$ hyperplanes in $\mathbb{R}^p$ is asymptotically bounded by

$$f_0^p(n^2 p^2) = \binom{p}{0}\binom{n^2 p^2}{p} \in \mathcal{O}(n^{2p} p^{2p}).$$

Hence, the size of $V(L_g)$ is bounded by $\mathcal{O}(n^{2p} p^{2p})$. Since there are $\mathcal{O}(m^p)$ possible vectors $g = (g_1, \ldots, g_p) \in \{1, \ldots, m\}^p$ with $g_1 \leq \cdots \leq g_p$, the cardinality of the FDS $V(L)$ is $\mathcal{O}(m^p n^{2p} p^{2p})$. As p is assumed to be fixed, this equals a size of $\mathcal{O}(m^p n^{2p})$.

Every subdivision L_g of $[0, 1]^p$ for a fixed g and all of its j-faces for $j = 1, \ldots, p$ can be constructed in $\mathcal{O}(n^{2p} p^{2p})$ time using the algorithm of Edelsbrunner et al. [9]. The evaluation of one candidate point can be realized in $\mathcal{O}(n(p + \log(n)))$ time. Therefore, all candidates in $V(L_g)$ need $\mathcal{O}(n^{2p+1} p^{2p}(p + \log(n)))$ time to be evaluated. As $\mathcal{O}(m^p)$ subdivisions have to be considered, the overall complexity of this approach can be bounded by $\mathcal{O}(m^p n^{2p+1} p^{2p}(p + \log(n)))$, which is mainly determined by the evaluation of the candidate points. With a fixed p, this simplifies to $\mathcal{O}(m^p n^{2p+1} \log(n))$ time for finding at least one optimal solution of the p-k-max problem utilizing the FDS $V(L)$. Note that for fixed values of k (not increasing with n) the kth largest entry can be determined more efficiently without a complete sorting, reducing the overall complexity to $\mathcal{O}(m^p n^{2p+1})$.

4.2. Linearity regions and objective value defining facilities

Using the fact that for $n-p \geq 2$ *every* optimal solution of the p-k-max problem has an objective value defining facility in the set EQ (see Theorem 3.3), the FDS $V(L)$ can be reduced significantly. Let z^* be the optimal objective function value and let

$$Y = \Big\{ x_1^* \in EQ_{ij}, i,j \in \{1,\dots,n\}, i \neq j \colon \exists\, X^* = \{x_1^*,\dots,x_p^*\} \in \mathcal{X}^* \\ \text{with } \ r_1 = d^w(v_i, x_1^*) = d^w(v_j, x_1^*) = z^* \Big\}$$

denote the set of all facility locations that are objective value defining facilities in at least one optimal solution in $\mathcal{X}^*$ (recall that r_1 denotes the radius of the cluster C_1 of customers associated to x_1^*). Since the FDS $\mathscr{C}^{p\geq 2}$ contains at least one optimal solution for each of the objective value defining facilities in the set Y, the set Y can be computed from the set $\mathcal{X}_{\mathscr{C}^{p\geq 2}}$ of optimal solutions resulting from an optimization process based on $\mathscr{C}^{p\geq 2}$ (see, for example, the recursive approach in [24]).

Now consider an arbitrary but fixed point $x_1^* = (e_{g_1^*}, t_1^*) \in Y$ on an edge $e_{g_1^*}$, i.e., $x_1^* \in EQ_{ij} \cap e_{g_1^*}$ for some $i,j \in \{1,\dots,n\}$, $i \neq j$. Then *all* alternative optimal solutions with x_1^* as objective value defining facility define the set

$$\mathcal{X}(x_1^*) = \left\{ X = \{x_1^*, x_2, \dots, x_p\} \colon x_2,\dots,x_p \in A(G) \ \wedge \ d^w(v_{\sigma(k)}, X) = z^* \right\},$$

and $\mathcal{X}^* = \bigcup_{x_1^* \in Y} \mathcal{X}(x_1^*)$. To guarantee that none of the optimal solutions in $\mathcal{X}(x_1^*)$ is missed, all p-tuples $g = (g_1^*, g_2, \dots, g_p)$ with $g_2 \leq \dots \leq g_p$ have to be enumerated. Note that the index g_1^* is fixed with x_1^* and kept in the first position of the p-tuple g to indicate that it corresponds to the objective value defining facility, even though $g_1^* \nleq g_2$ in general. Since the assumption $g_1 \leq g_2 \leq \dots \leq g_p$ was used only to avoid duplications, this has no effect on the correctness of the following analysis.

Given an arbitrary but fixed p-tuple $g = (g_1^*, g_2, \dots, g_p)$ with $g_2 \leq \dots \leq g_p$, all alternative optimal solutions in $\mathcal{X}(x_1^*)$ have to lie on that equilibrium hyperplane in $[0,1]^p$ that corresponds to $B_{ij11}^{\alpha,\beta}$ and that is given by $\{(t_1,\dots,t_q)^T \in \mathbb{R}^p : t_1 = t_1^*\} =: eq_g(x_1^*)$. Let $L_g(x_1^*)$ be the subdivision of $[0,1]^p$ arising from L_g by deleting all bottleneck hyperplanes of type $B_{cc11}^{\alpha\beta}$ and all equilibrium hyperplanes of type $B_{cd11}^{\alpha\beta}$ except for $eq_g(x_1^*)$ for all $c,d = 1,\dots,n$, $c \neq d$, and $\alpha, \beta \in \{+,-\}$. Note that these hyperplanes are parallel to $eq_g(x_1^*)$ and contain therefore no solution in $\mathcal{X}(x_1^*)$. $C(L_g(x_1^*))$ denotes the set of cells of this subdivision. Moreover, let

$$L_g^{eq}(x_1^*) := L_g(x_1^*) \cap eq_g(x_1^*)$$

be the subdivision $L_g(x_1^*)$ restricted to $eq_g(x_1^*)$, and let $V(L_g^{eq}(x_1^*))$ be the set of all 0-faces of the subdivision $L_g(x_1^*)$ that lie on the hyperplane $eq_g(x_1^*)$, i.e., the set of all points on $eq_g(x_1^*)$ that are intersected by at least $p-1$ of the other hyperplanes in $L_g(x_1^*)$.

Corollary 4.6. *Let $x_1^* = (e_{g_1^*}, t_1^*) \in Y$ and let $g = (g_1^*, g_2, \dots, g_p)$ be an arbitrary but fixed vector of edge indices with $g_2 \leq \dots \leq g_p$. Then the k-*max *function is linear on every intersection of a cell of $C(L_g(x_1^*))$ with $eq_g(x_1^*)$.*

Proof. Follows by construction and using Theorem 4.3. □

Corollary 4.6 immediately implies the following result.

Corollary 4.7. *Let $X^* = \{x_1^*, \dots, x_p^*\} \in \mathcal{X}_{\mathscr{C}^{p\geq 2}}$ with $x_1^* = (e_{g_1^*}, t_1^*) \in Y$ and $g_1^* \in \{1,\dots,m\}$. If the p-k-*max *problem with $n, p \geq 2$ and $k \leq n-p$ has more than one optimal solution in $\mathcal{X}(x_1^*)$, at least one of these alternative optimal solutions can be found in the set*

$$V(L^{eq}(x_1^*)) := \bigcup_{\substack{g=(g_1^*,g_2,\dots,g_p)\in\{1,\dots,m\}^p \\ g_2 \leq \dots \leq g_p}} V(L_g^{eq}(x_1^*)).$$

Taking the union over all candidates $x_1^* \in Y$, an extended FDS is obtained as

$$V(L^{eq}(Y)) := \bigcup_{x_1^* \in Y} V(L^{eq}(x_1^*)).$$

Note that by construction, $(\mathscr{C}^{p\geq 2} \cap \mathcal{X}^*) \subseteq V(L^{eq}(Y))$, and that in general both sets do not have to be equal since the arrangement of hyperplanes generating $V(L^{eq}(Y))$ also includes bottleneck and balance hyperplanes. Moreover, since for every $x_1^* \in Y$, the k-max function is linear on every intersection of a cell in $C(L_g(x_1^*))$ with $eq_g(x_1^*)$ (see Corollary 4.6), the optimal solutions in the FDS $V(L^{eq}(Y))$ can be used as seed points for the generation of the *complete* optimal set $\mathcal{X}^*$. This is the topic of Section 4.3.

To summarize, the procedure to determine all those optimal solutions of the p-k-max problem that lie in the FDS $V(L^{eq}(Y))$ (and hence all seedpoints needed for the determination of the complete optimal set) is given by the following steps: For a fixed $x_1^* \in Y$, at first, the subdivisions $L_g(x_1^*)$ of $[0,1]^p$ for all $g = (g_1^*, g_2, \ldots, g_p) \in \{1, \ldots, m\}^p$, $g_2 \leq \cdots \leq g_p$, have to be constructed. Afterwards, the 0-faces, i.e., the intersection points of $eq_g(x_1^*)$ with the other relevant equilibrium and balance hyperplanes, have to be determined. Then, the k-max function is evaluated in these candidate points, and all such points leading to the smallest objective function value z^* are optimal. This is done for all objective value defining facilities in $x_1^* \in Y$, which can be obtained in a preprocessing procedure based on the set $\mathscr{C}^{p\geq 2}$, see [24]. The procedure is described in more detail in Algorithm 1.

The computational effort of considering all elements of the set Y is $\mathcal{O}(mn^2)$, which is the maximum number of equilibrium points of G. For fixed $x_1^* \in Y$, the number of p-tuples $g = (g_1, \ldots, g_p) \in \{1, \ldots, m\}^p$ that have to be considered is $\mathcal{O}(m^{p-1})$ since $g_1 = g_1^*$ is fixed. The maximum number of equilibrium-, bottleneck- and balance hyperplanes is $|H| = \mathcal{O}(n^2p^2)$. For the computation of H' in Step 9, all these hyperplanes have to be considered and each intersection of $h \in H$ and $eq_g(x_1^*)$ can be determined in $\mathcal{O}(p)$ time. The subdivision $L_g^{eq}(x_1^*)$ can be constructed with the algorithm of Edelsbrunner [8] in $\mathcal{O}((n^2p^2)^{p-1})$ time as $L_g^{eq}(x_1^*)$ is a subdivision of the unit hypercube $[0,1]^{p-1}$. Following [8], the number of 0-faces in $V(L_g^{eq}(x_1^*))$ is bounded by $\mathcal{O}((n^2p^2)^{p-1})$. As every candidate solution needs $\mathcal{O}(n(p + \log(n)))$ time to be evaluated, the evaluation of all 0-faces in $L_g^{eq}(x_1^*)$ has a complexity of $\mathcal{O}(n^{2p-1}p^{2p-2}(p + \log(n)))$. Hence, an overall complexity for Algorithm 1 of $\mathcal{O}(m^p n^{2p+1} \log(n))$ (for fixed p) follows, which is mainly determined by the evaluation of the FDS $V(L^{eq}(Y))$.

Algorithm 1 Local Analysis of the p-k-max problem on networks

---

Input: Graph $G = (V_G, E)$; customers $V = \{v_1, \ldots, v_n\} \subseteq V_G$ with $w_i > 0$ for all $i = 1, \ldots, n$; $p \in \{2, \ldots, n\}$; $k \in \{1, \ldots, n-p\}$; optimal objective value z^*; set Y of z^*-defining equilibrium points.

1: $V_{opt} := \emptyset$; $H' := \emptyset$
2: **for all** $x_1^* = (e_{g_1^*}, t_1^*) \in Y$ **do** *// z^*-defining facilities*
3:   **for all** $(g_2, \ldots, g_p)^T \in \{1, \ldots, m\}^{p-1}$: $g_2 \leq \ldots \leq g_p$ **do**
4:     $g := (g_1^*, g_2, \ldots, g_p)$ *// p-tuples g with fixed edge g_1^**
5:     Derive the set H of equilibrium-, bottleneck- and balance hyperplanes for $e_{g_1^*} \times \ldots \times e_{g_p}$
6:     **for all** $a = 1, \ldots, |H|$ **do**
7:       $h' = h_a \cap eq_g(x_1^*)$ with $h_a \in H$
8:       $H' = H' \cup h'$ *// Hyperplanes defining the subdivision restricted to $eq_g(x_1^*)$*
9:     Derive the subdivision $L_g^{eq}(x_1^*) = L_g(x_1^*) \cap eq_g(x_1^*)$ of H'
10:     **for all** $X_i \in V(L_g^{eq}(x_1^*))$ **do** *// Test all 0-faces for optimality*
11:       $z_i = k\text{-max}(d^w(V, X_i))$ *// Objective function value*
12:       **if** $z_i = z^*$ **then** *// X_i is optimal*
13:         $V_{opt} = V_{opt} \cup \{X_i\}$

Output: Set $\mathcal{X} = V_{opt}$ of optimal solutions of the p-k-max problem with optimal objective function value z^*

Note that the set $V(L^{eq}(Y))$ contains alternative optimal solutions (if existent) that cannot be found in the candidate set $\mathscr{C}^{p\geq 2}$. Thus, the application of Algorithm 1 provides useful additional information without increasing the overall worst case complexity determined by the evaluation of the FDS $\mathscr{C}^{p\geq 2}$. Note also that the algorithm can be parallelized easily. Preliminary computational tests indicate that this leads to a significant improvement of the computation time in practice.

Remark 4.8. The worst case complexity of $\mathcal{O}(m^p n^{2p+1} \log(n))$ is generally a very pessimistic estimate as the number of optimal 0-faces as well as the number of all other relevant s-faces for $s \in \{1, \ldots, p-1\}$ is upper bounded using the maximum possible number of faces. However, in practice, the number of these faces is in general much smaller, see Section 5 for corresponding computational results.

4.3. Determining all (infinitely many) optimal solutions

There may still be optimal solutions that cannot be found in the FDS $V(L^{eq}(Y))$. In Example 4.1, it is easy to verify that not only the two points X and $\bar{X}$ are optimal: All alternative optimal solutions $\hat{X} = \{x_1, \hat{x}_2\}$ with objective value defining facility $x_1 = (e_{34}, \frac{1}{3})$ are given by

$$\hat{x}_2 \in \left[\left(e_{12}, 0\right), \left(e_{12}, \frac{1}{3}\right)\right] \cup \left[\left(e_{13}, 0\right), \left(e_{13}, \frac{1}{3}\right)\right] \cup \left[\left(e_{15}, 0\right), \left(e_{15}, \frac{2}{3}\right)\right].$$

Please cite this article as: T. Schnepper, K. Klamroth, J. Puerto et al., A local analysis to determine all optimal solutions of p-k-max location problems on networks, Discrete Applied Mathematics (2020), https://doi.org/10.1016/j.dam.2020.05.013.

These additional optimal solutions can be easily computed when the optimal solutions in $V(L^{eq}(x_1^*))$ are known. For this purpose, let conv(A) be the convex hull of the elements in a set A.

Theorem 4.9. *Let $n, p \geq 2$ and $k \leq n-p$, and let $X^* = \{x_1^*, \dots, x_p^*\} \in \mathcal{X}_{\mathscr{C}^{p\geq 2}}$ with optimal objective function value z^* and $x_1^* = (e_{g_1^*}, t_1^*) \in Y$ with $g_1^* \in \{1, \dots, m\}$ be fixed. Moreover, for a fixed $g = (g_1^*, g_2, \dots, g_p) \in \{1, \dots, m\}^p$, $g_2 \leq \dots \leq g_p$, and a cell $\bar{C} \in C(L_g(x_1^*))$, let*

$$V^*(\bar{C}^{eq}) := \left\{\bar{X} \in V(L_g^{eq}(x_1^*)) \;:\; \bar{X} \in \bar{C} \;\wedge\; d^w(V, \bar{X}) = z^*\right\}$$

be the set of all vertices of the cell $\bar{C}$ that are in $eq_g(x_1^*)$ and that are optimal for the p-k-max problem. Then, all solutions

$$X = \{x_1^*, x_2, \dots, x_p\} \quad \text{with} \quad X \in \text{conv}(V^*(\bar{C}^{eq}))$$

are optimal solutions of the p-k-max problem.

Proof. Recall from Corollary 4.6 that the k-max function is linear on every intersection of a cell $\bar{C} \in C(L_g(x_1^*))$ with $eq_g(x_1^*)$. Let $\bar{C}^{eq} = \bar{C} \cap eq_g(x_1^*)$. Since all solutions in $V^*(\bar{C}^{eq})$ have the same (optimal) objective value z^* and since conv$(V^*(\bar{C}^{eq})) \subseteq \bar{C}^{eq}$, this implies that the k-max function is constant over conv$(V^*(\bar{C}^{eq}))$. Consequently, all solutions $X = \{x_1^*, x_2, \dots, x_p\}$ with $X \in \text{conv}(V^*(\bar{C}^{eq}))$ are optimal solutions of the p-k-max problem. □

As a consequence of Theorem 4.9, b-faces up to a dimension of $b = p-1$ can be optimal for the p-k-max problem. A whole cell of the subdivision cannot be optimal because otherwise the objective value defining facility would not be an equilibrium point. Thus, the complete set of optimal solutions $\mathcal{X}^*$ can be determined by enumerating all such b-faces.

From an algorithmic perspective, if only one new facility is to be located, i.e., $p = 1$, then this facility is always objective value defining. In this case, the linearity regions of the k-max function can be derived from the set EQ, see [24]. All optimal solutions can then be easily computed. In the special case of two new facilities, determining the set $V^*(\bar{C}^{eq})$ of all optimal intersection points that belong to the same cell is still quite easy since two optimal, adjacent 0-faces can be identified by sorting the coordinates of the optimal intersection points with respect to their x_2-coordinate.

Example 4.10 (*Continuation of Example 4.1*)**.** The local analysis is applied to the 2-1-max problem on the graph introduced in Example 4.1. An optimal solution computed by the evaluation of the FDS $\mathscr{C}^{p\geq 2}$ (see Theorem 3.3) is

$$X = \{x_1, x_2\} \quad \text{with} \quad x_1 = \left(e_{34}, \frac{1}{3}\right) \quad \text{and} \quad x_2 = \left(e_{12}, \frac{1}{3}\right),$$

with optimal objective value $z = \frac{4}{3}$ and objective value defining facility x_1. Since $x_1 \in e_6 = e_{34}$, this determines $g_1 = 6$. The arrangement of hyperplanes $L_{(6,h)}(x_1)$ for $h \in \{1, \dots, 5, 7\}$ are shown in Fig. 7. Note that in the case of $p = 2$ new facilities, and given that $k \leq n-p$, the two new facilities are on different edges in all optimal solutions (see [23]). Thus, the case $L_{(6,6)}(x_1)$ is not considered. The analyzed candidate intersection points are marked with an empty circle, the intersection points and the line segments that contain the local minima are marked with filled circles in gray (see Fig. 7(d)–(f)). The local minima that are also globally optimal are drawn in black (see Fig. 7(a)–(c)). Therefore, the alternative optimal solutions with x_1 as objective value defining facility are given by $\mathcal{X}(x_1) = \{x_1, \tilde{x}_2\}$ with

$$\tilde{x}_2 \in \left[\left(e_{12}, 0\right), \left(e_{12}, \frac{1}{3}\right)\right] \cup \left[\left(e_{13}, 0\right), \left(e_{13}, \frac{1}{3}\right)\right] \cup \left[\left(e_{15}, 0\right), \left(e_{15}, \frac{2}{3}\right)\right].$$

These alternative optimal solutions are illustrated in Fig. 8.

For the case $p \geq 3$ this identification is much more complicated. An outline of a possible implementation is given in the following. Edelsbrunner [8] introduced an algorithm for constructing an arrangement of hyperplanes in $\mathbb{R}^p$, i.e., an algorithm that builds up a data structure $I(H) = (V(I), E(I))$ called *incidence graph* which stores all faces of the arrangement and also all incidences between pairs of faces. Let H be a set of $|H| = a$ hyperplanes in $\mathbb{R}^p$ and let $A(H)$ be the arrangement resulting from H. Each face f of $A(H)$ is represented by a vertex $v(f) \in V(I)$ and if two faces f and f' are incident, the vertices $v(f)$ and $v(f')$ are connected by an edge. Besides the regular s-faces of dimension $0 \leq s \leq p$, two more faces are defined: The (-1)-face (representing an empty set) is incident with all vertices of $I(H)$ representing a 0-face and the $(p+1)$-face (representing $A(H)$) is incident with all vertices representing a p-face (see Fig. 9). Note that an incidence graph of a set of a hyperplanes in $\mathbb{R}^p$ contains $\mathcal{O}(a^p)$ vertices and edges.

The incidence graph can be used to analyze which optimal 0-faces of $L_g(x_1^*)$ belong to the same cell, implying that their convex hull is optimal for the p-k-max problem. The subdivision of $[0,1]^p$ induced by $L_g(x_1^*)$ for a fixed p-tuple of edges g is an arrangement of at most $\mathcal{O}(n^2p^2)$ hyperplanes. As x_1^* is fixed, and with it also the equilibrium hyperplane $eq_g(x_1^*)$, it is sufficient to analyze the subdivision $L_g^{eq}(x_1^*) = L_g(x_1^*) \cap eq_g(x_1^*)$ in $\mathbb{R}^{p-1}$. The notation for the faces (vertices, cells etc.) of $L_g^{eq}(x_1^*)$ are adapted from the subdivisions before. Let $I(L_g^{eq}) = (V(I), E(I))$ denote the incidence graph of $L_g^{eq}(x_1^*)$. The idea is to go bottom-up through the incidence graph and to identify for each dimension $s \in \{0, \dots, p-1\}$ the optimal s-faces. Note that the optimal 0-faces can be determined easily. An s-face for $s \geq 2$ is only optimal if all of its incident $(s-1)$-faces

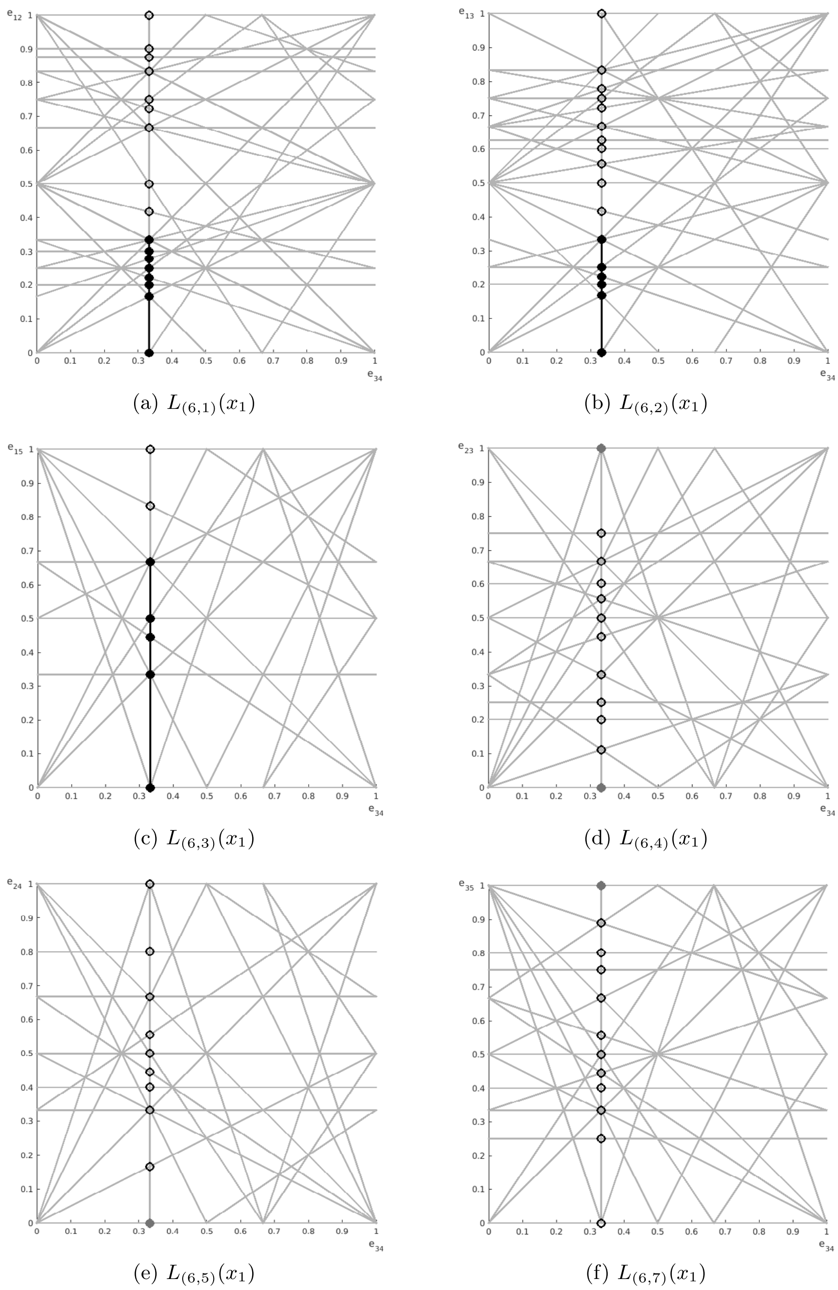


Fig. 7. Arrangement of hyperplanes $L_{(6,h)}(x_1)$ with $h = 1, \ldots, 7$, $h \neq 6$. Intersection points (circles), local minima (gray dots) and global minima (black dots). The latter are the optimal solutions in $\mathcal{X} = \mathcal{X}(x_1)$ of the 2-1-max problem.

are optimal. Thus, let

$$V_s := \left\{ v_i^s \in V(I) : f_i^s \text{ is } i\text{th optimal } s\text{-face}, i \in \{1, \ldots, |V_s|\} \right\}$$

with $s \in \{0, \ldots, p-1\}$. Note that all $X \in f_i^s$ for a corresponding $v_i^s \in V_s$ are optimal solutions for the p-k-max problem and that V_0 equals the set of optimal extreme points in $V(L_g^{eq}(x_1^*))$. In the following, no distinction will be made between the faces f_i^s of $L_g^{eq}(x_1^*)$ and the nodes $v_i^s \in V(I)$ that represent them.

It is now assumed that the set V_{s-1} of optimal $(s-1)$-faces is known. Let $v_i^{s-1} \in V_{s-1}$ be the ith optimal $(s-1)$-face, $i \in \{1, \ldots, |V_{s-1}|\}$, and let v_j^s be an s-face incident to v_i^{s-1} such that $(v_i^{s-1}, v_j^s) \in E(I)$. Then, v_j^s is optimal for the p-k-max

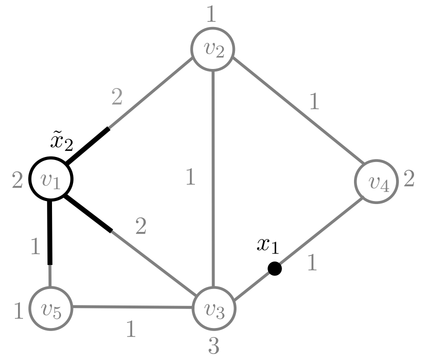


Fig. 8. Set $\mathcal{X}$ (black) of all optimal solutions of the 2-1-max problem with $z^* = \frac{4}{3}$.

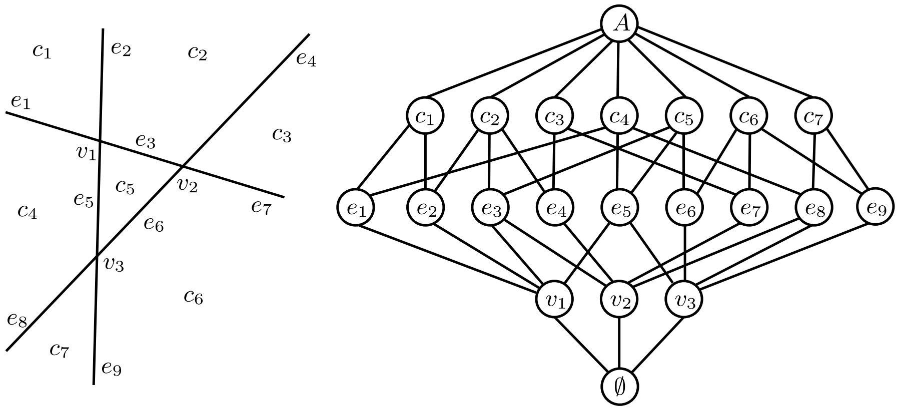


Fig. 9. Arrangement $A(H)$ in $\mathbb{R}^2$ and its incidence graph $I(H)$.

problem if and only if it holds that

$$v_h^{s-1} \in V_{s-1} \quad \text{for all} \quad (v_h^{s-1}, v_j^s) \in E \quad \text{with} \quad h \in \{1, \dots, |V_{s-1}|\}, \tag{17}$$

i.e., if all of its incident subfaces v_h^{s-1} are optimal. Note that it is enough to analyze just one incident optimal $v_{i'}^s \neq v_i^{s-1}$ because then the other incident subfaces also have to be optimal as the k-max function is constant over f_j^s if it is constant over two of its $(s-1)$-dimensional boundary faces. Thus, if condition (17) is satisfied, then v_j^s is an element of V_s. If condition (17) is not satisfied for all superfaces f_j^s of f_i^{s-1}, then f_i^{s-1} does not contribute to an optimal face of a larger dimension and is stored in the set F_{opt} of optimal faces of $L_g^{eq}(x_1^*)$. As a consequence, only optimal faces of maximum dimension are stored and not also all their smaller-dimensional subfaces.

As at least $s+1$ optimal $(s-1)$-faces are needed to construct an optimal s-face, the condition $|V_{s-1}| < s+1$ is used as a stopping criterion. A set EP_{opt} is generated analogously to F_{opt}. Each element M of EP_{opt} is a set that contains the optimal 0-faces of a cell $\bar{C} \in C(L_g^{eq}(x_1^*))$ having at least one optimal 0-face.

Going through all dimensions $s \in \{0, \dots, p-1\}$ needs at most $\mathcal{O}(p)$ time. The number of all $(s-1)$-faces is bounded by $\mathcal{O}((n^2p^2)^{p-1})$ (see [8]). Thus, $|V_{s-1}| = \mathcal{O}((n^2p^2)^{p-1})$. All s-faces v_j^s have to be considered, not only the optimal ones, which leads to $\mathcal{O}((n^2p^2)^{p-1})$ time. To find all subfaces v_i^{s-1} incident to v_j^s, all predecessors of v_j^s have to be analyzed. Since a face can be bounded by at most all n^2p^2 hyperplanes, the complexity to check this is $\mathcal{O}(n^2p^2)$. Considering all $(s-1)$-faces that are incident to v_j^s also needs $\mathcal{O}(n^2p^2)$ time in the worst case. This leads to a complexity of $\mathcal{O}(n^{4p}p^{4p+1})$ resp. $\mathcal{O}(n^{4p})$ for fixed p. In practice this procedure can be expected to be more efficient as the number of optimal s-faces is in general much smaller than the total number of s-faces.

5. Computational results

In the following, the local analysis approaches for the p-k-max problem developed in the previous chapters are compared and computationally evaluated for the case of two new facilities, i.e. $p = 2$. Thereby, two versions of the local analysis are compared: The evaluation of the FDS $V(L')$ (see Theorem 4.4), in the following referred to as *full local analysis*, and the evaluation of the reduced FDS $V(L^{eq}(Y))$ (see Corollary 4.7), named *reduced local analysis*.

Table 1
Evaluation of the FDS $V(L')$ (see Theorem 4.4): Numbers of candidates and CPU-times in seconds for given number of n customers and given density ρ of the graph, distinguished w.r.t. the value of k.

n	ρ	$\lvert V(L')\rvert$	$k = 1$	$k = 0.25n$	$k = 0.75n$
10	0.1	–	–	–	–
	0.3	1 043 016	11.47	11.49	11.49
	0.5	4 667 700	50.94	51.02	50.99
20	0.1	11 783 328	200.34	200.59	200.28
	0.3	453 873 031	8029.16	8032.07	8069.94
	0.5	>900 000 000	>18 000	>18 000	>18 000
30	0.1	291 740 500	7883.21	7904.61	7903.58
	0.3	>700 000 000	>18 000	>18 000	>18 000

Table 2
Evaluation of the reduced local analysis (Corollary 4.7): Numbers of candidates and CPU-times in seconds for given number of n customers and given density ρ of the graph, both distinguished w.r.t. the value of k.

n	ρ	$k = 1$		$k = 2$		$k = 0.25n$		$k = 0.5n$		$k = 0.75n$	
		$\lvert V(L^{eq}(Y))\rvert$	Time [s]	$\lvert V(L^{eq}(Y))\rvert$	Time [s]	$\lvert V(L^{eq}(Y))\rvert$	Time [s]	$\lvert V(L^{eq}(Y))\rvert$	Time [s]	$\lvert V(L^{eq}(Y))\rvert$	Time [s]
10	0.1	–	–	–	–	–	–	–	–	–	–
	0.3	1 443	0.205	1 447	0.164	1 495	0.136	1 499	0.075	1 467	0.069
	0.5	3 200	0.622	3 242	0.461	3 315	0.338	3 315	0.174	3 173	0.073
20	0.1	5 122	1.371	5 079	1.260	5 035	0.978	5 044	0.465	5 033	0.180
	0.3	30 056	44.915	30 351	35.598	32 559	22.119	31 904	8.813	31 799	1.815
	0.5	59 202	153.727	58 902	115.934	59 796	64.449	60 737	24.503	60 283	3.858
30	0.1	33 344	60.932	33 389	53.101	33 227	29.416	33 273	13.567	33 581	2.924
	0.3	164 714	1411.676	172 522	1140.710	175 160	503.864	174 677	211.478	175 079	31.636
	0.5	305 911	4502.882	306 436	3762.006	314 682	1680.585	314 009	639.868	302 514	73.939
50	0.1	363 792	7135.507	355 796	6196.555	349 921	2495.658	355 217	949.539	362 752	212.869
	0.3	–	>18 000	–	>18 000	–	>18 000	–	>18 000	1 425 648	2220.202
	0.5	–	>18 000	–	>18 000	–	>18 000	–	>18 000	2 569 633	7684.503

All numerical tests are performed on a compute server with 4 Intel Xeon E7540 Hexacore (2.0 GHz) and 128 GB RAM using a single thread. All algorithms are implemented and run in MATLAB, version R2013a. They were tested on randomly generated Euclidean graphs: Given a number n of nodes and a density ρ of the graph, first the customer nodes are randomly placed in integer coordinates $v_i = (x_1^i, x_2^i) \in \mathbb{Z}^2$ in the plane (normally distributed with mean 50 and standard deviation of 30. The associated weights are randomly chosen from the set $w_i \in \{1, \dots, 15\}$, $i = 1, \dots, n$. Different values of k between $k = 1$ and $k = 0.75n$ are distinguished, while the number of new facilities to locate remains fixed to $p = 2$. All tests are conducted up to a maximum computation time of five hours.

5.1. Full local analysis: Construction and evaluation of $V(L')$

Table 1 summarizes the sizes of the FDS $V(L')$ for the tested instances and the corresponding computation times for its evaluation. The size of the candidate set (and therefore also the CPU-time) increases dramatically with the size of the underlying problem. As expected, the number of candidates and the computation times do not depend on the value of k as the constructed subdivisions are independent of k. Moreover, the cardinality of $V(L')$ is significantly larger than the cardinality of $\mathscr{C}^{p\geq 2}$: $\mathscr{C}^{p\geq 2}$ contains on average only 1.86% of the candidates in comparison to $V(L')$. However, the evaluation of the FDS $V(L')$ contains in general a larger set of optimal solutions of the problem. Since the same solution sets can be found with the reduced local analysis based on the set $V(L^{eq}(Y))$, this is evaluated in the following section.

5.2. Reduced local analysis: Construction and evaluation of $V(L^{eq}(Y))$

The results of the tests for Algorithm 1 are given in Table 2. Note that the discrete version of the algorithm is tested, i.e., not all (infinitely many) optimal solutions are computed but only the solutions belonging to the FDS $V(L^{eq}(Y))$. From these, the complete optimal set can be easily constructed. The number of candidates gives the number of intersection points for all subdivisions. The CPU-time is measured for the overall procedure, including the recursive approach from [24] to construct the set Y.

Note that, as the cardinality of the set Y of objective function value defining facilities may vary w.r.t. the value of k, also the total number of candidates $\lvert V(L^{eq}(Y))\rvert$ may differ with varying values of k. The computation times, in contrast, decrease with an increasing value of the parameter k since the CPU-times of the recursive approach depend on k. It should be mentioned that the given CPU-times are mainly determined by the recursive approach. The local analysis itself takes only 14.38% on average of the overall computation times. Hence, the local analysis computes in general many alternative optimal solutions with only small additional effort.

Table 3
Number of optimal solutions in $V(L^{eq}(Y))$ determined with the reduced local analysis.

n	ρ	$k = 1$	$k = 2$	$k = 0.25n$	$k = 0.5n$	$k = 0.75n$
10	0.3	116.70	101.60	89.20	78.25	310.55
	0.5	85.80	99.85	98.70	137.00	512.20
20	0.1	620.15	415.25	300.85	247.60	190.55
	0.3	378.25	499.35	152.10	445.60	353.15
	0.5	533.55	690.90	233.45	537.95	987.45
30	0.1	1353.90	484.15	594.55	268.55	301.10
	0.3	709.45	178.75	737.65	651.50	1051.50
	0.5	1624.20	670.30	398.20	696.55	1378.30

5.3. Comparison of the full and the reduced local analysis

Obviously, the reduced local analysis yields much better results in terms of the CPU-time for all test instances than the full local analysis. The reduced analysis needs on average over all test problems only 0.054% of the intersection points. This does not vary significantly for different values of k. Moreover, the reduced local analysis needs only 1% of the CPU-times as compared to the full local analysis for $k = 1$, 0.60% for $k = 0.25n$, 0.30% for $k = 0.5n$ and 0.18% for $k = 0.75n$. This can be explained by the corresponding dependence of the recursive approach on the parameter k, see also [24].

5.4. Complementing objectives: Selection of a most preferred solution

Optimization problems with bottleneck objectives (like p-k-max location problems) often have many or even infinitely many optimal solutions. Even though these alternative optimal solutions share the same p-k-max objective function value, they can be of completely different structure. Due to the computation of the complete set of optimal solutions in a local analysis, secondary, possibly complementing objective functions can be used in a decision making step to select to most preferred solution. This approach can be considered as the lexicographic optimization of a biobjective optimization problem with k-max being the first objective. Then among all optimal solution wrt. the k-max-objective a secondary objective function is optimized [10]. To illustrate the potential of secondary objective functions for decision making, we evaluate the finite set of alternative optimal solutions (for each problem size on 20 randomly chosen instances) obtained with the local analysis (cf. Table 3). In a similar way the results can be extended to the complete set of optimal solutions.

Depending on the considered application different secondary objectives can be relevant. The following evaluation is restricted to three classical objective functions to show the potential gain in solution quality for the non-outlier facilities, which we denote in the following as inliers $V \setminus V^*_{k-1}$: (a) The average Weber objective contribution of the inliers $\bar{z}_W := \frac{z_W(V\setminus V^*_{k-1})}{|V\setminus V^*_{k-1}|}$ with $z_W(V \setminus V^*_{k-1}) := \sum_{v\in V\setminus V^*_{k-1}} d^w(v, X)$, (b) the number $|DC|$ of customers with double coverage within the coverage radius z^*, and (c) the size of the largest cluster in relation to the number of inliers $\frac{|C_{\max}|}{n-k+1}$. Note that in the evaluation of the Weber objective function (a), outliers are excluded to obtain a measure on how compact the majority of inlier customers lie inside their respective clusters. The motivation for quality measure (b) is that double covered customers can be considered as more robustly supplied. Criterion (c) relates to capacity constraints, however, even in the case of uncapacitated facility location as considered in this paper it is preferable to allocate customers uniformly to new facilities. If $C_{\max}$ denotes the cluster covering the largest number of customers, then $\frac{|C_{\max}|}{n-k+1}$ is the relative size of the largest cluster related to the total number of inliers.

For these three quality measures, the maximum and minimum value over the set of k-max-optimal solutions are determined and averaged over 20 instances for each problem type with $n = 20$ customers. Table 4 illustrates how different alternative k-max-optimal solutions perform w.r.t. secondary objective functions. Interestingly, both the average contribution to the Weber objective function of each inlier $\bar{z}_W$ and the number of double covered customers $|DC|$ reduce clearly with increasing values of k. This is due to the fact that for large numbers of k only a small percentage of the customers is covered, and that the new facilities are located as centers of compact clusters which might be far away from each other. Moreover, the average contribution to the Weber objective function $\bar{z}_W$ decreases with increasing values of ρ since the average distance between customers are smaller in denser graphs. For the relative size of the largest cluster, no clear trend depending on ρ and k can be observed.

For all three secondary objective functions considered above, the improvements that are possible when the complete optimal set is known (and hence a most preferred solution can be selected) are significant. Since the presented local analysis comes with comparably small additional computational effort, this is a clear statement in favor of computing alternative optimal solutions.

6. Conclusions

In this paper an efficient approach for determining *all* optimal solutions of the p-k-max problem on a graph was introduced. Starting from seed points from a finite dominating set, alternative optimal solutions of the p-k-max problem

Table 4
Evaluation of alternative optimal solutions of the *k-max* problem with $n = 20$ customers w.r.t. Weber objective z_W, number of double covered customers $|\mathrm{DC}|$, and size of the largest cluster in relation to the number of inliers $\frac{|C_{\max}|}{n-k+1}$.

k	ρ	$\min \bar{z}_W$	$\max \bar{z}_W$	$\min \|\mathrm{DC}\|$	$\max \|\mathrm{DC}\|$	$\min \frac{\|C_{\max}\|}{n-k+1}$	$\max \frac{\|C_{\max}\|}{n-k+1}$
1	0.1	157.559	182.797	7.30	11.55	0.670	0.733
	0.3	77.646	87.941	8.55	9.60	0.608	0.685
	0.5	71.450	84.825	7.55	9.55	0.618	0.695
2	0.1	130.541	149.727	5.75	8.20	0.666	0.721
	0.3	69.950	78.032	6.45	8.10	0.618	0.692
	0.5	60.270	68.301	5.95	7.35	0.613	0.676
5	0.1	94.561	106.986	3.30	4.85	0.662	0.703
	0.3	54.139	57.724	4.15	4.50	0.594	0.625
	0.5	38.693	44.834	3.80	4.60	0.575	0.616
10	0.1	49.293	57.312	0.75	1.10	0.586	0.623
	0.3	26.383	30.886	1.65	2.15	0.559	0.623
	0.5	23.108	27.349	1.90	2.25	0.541	0.605
15	0.1	19.705	25.721	0.00	0.10	0.508	0.558
	0.3	9.368	12.559	0.20	0.25	0.475	0.508
	0.5	7.860	11.826	0.05	0.20	0.525	0.583

were obtained by performing a local analysis on each p-tuple of edges possibly containing the p new facilities. More precisely, for a fixed p-tuple of edges all feasible solutions were associated with points in the unit hypercube $[0, 1]^p$. It was shown that the k-max function is piecewise linear and concave on every cell of a subdivision of $[0, 1]^p$ obtained from an arrangement of so-called equilibrium hyperplanes and balance hyperplanes. Thus, the 0-faces of the arrangement of hyperplanes are a finite dominating set for the p-k-max problem. For a fixed value of p, the finite dominating set is of polynomial size and the p-k-max problem can be solved in polynomial time.

This candidate set is further reduced using the information given by the set of optimal objective value defining facilities. As a consequence, only those 0-faces lying in a specific hyperplane of the subdivisions are needed. This reduces the number of candidates enormously. Computational tests underlined this improvement in terms of much smaller CPU-times. Moreover, all (infinitely many) optimal solutions of the p-k-max problem can be obtained by constructing the convex hull of every set of optimal 0-faces that belong to the same cell of a subdivision. Having access to the complete optimal set of p-k-max problems paves the ground for considering secondary objective functions and thus selecting a most preferred solution from the often large set of solution alternatives. Computational tests were performed with averaging criteria (e.g., w.r.t. total cost or equally distributed capacities) and robustness considerations (e.g., double coverage). The results showed significant differences in the performance of solution alternatives. This is a strong indication that computing complete optimal sets in center type location problems opens up an important and often unexplored optimization potential.

CRediT authorship contribution statement

Teresa Schnepper: Conceptualization, Methodology, Formal analysis, Software, Writing - original draft, Writing - review & editing. **Kathrin Klamroth:** Conceptualization, Methodology, Formal analysis, Writing - original draft, Writing - review & editing. **Justo Puerto:** Conceptualization, Methodology, Formal analysis, Writing - original draft, Writing - review & editing. **Michael Stiglmayr:** Methodology, Formal analysis, Software, Writing - original draft, Writing - review & editing.

Acknowledgments

Teresa Schnepper was partially supported by the Foundation of German Business (sdw). Justo Puerto is partially supported by Spanish Ministerio de Economía y Competitividad grant number MTM2016-74983-C02-01. Furthermore, we would like to thank the anonymous reviewer for the profound and very constructive comments, which helped us to significantly improve this article.

References

[1] P. Agarwal, J. Phillips, An efficient algorithm for 2D euclidean 2-center with outliers, in: 16th Annual European Symposium on Algorithms (ESA), 2008, pp. 64–75.

[2] H.-K. Ahn, S. Bae, E. Demaine, M. Demaine, S.-S. Kim, M. Korman, I. Reinbacher, W. Son, Covering points by disjoint boxes with outliers, Comput. Geom. 44 (3) (2011) 178–190, http://dx.doi.org/10.1016/j.comgeo.2010.10.002.

[3] R.E. Burkard, F. Rendl, Lexicographic bottleneck problems, Oper. Res. Lett. 10 (1991) 303–308, http://dx.doi.org/10.1016/0167-6377(91)90018-k.

[4] E. Carrizosa, F. Plastria, On minquantile and maxcovering optimisation, Math. Program. 71 (1) (1995) 101–112, http://dx.doi.org/10.1007/bf01592247.

[5] E. Carrizosa, F. Plastria, Polynomial algorithms for parametric minquantile and maxcovering planar location problems with locational constraints, Top 6 (2) (1998) 179–194, http://dx.doi.org/10.1007/bf02564786.

[6] D. Chakrabarty, P. Goyal, R. Krishnaswamy, The non-uniform k-center problem, 2016, CoRR abs/1605.03692. URL: http://arxiv.org/abs/1605.03692, arXiv:1605.03692.
[7] M. Charikar, S. Khuller, D. Mount, G. Narasimhan, Algorithms for facility location problems with outliers, in: Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001, pp. 642–651, URL: http://dl.acm.org/citation.cfm?id=365411.365555.
[8] H. Edelsbrunner, Algorithms in Combinatorial Geometry, Springer Berlin Heidelberg, 1987.
[9] H. Edelsbrunner, J. O'Rourke, R. Seidel, Constructing arrangements of lines and hyperplanes with applications, SIAM J. Comput. 15 (1986) 341–363, http://dx.doi.org/10.1137/0215024.
[10] M. Ehrgott, Multicriteria Optimization, second ed., Springer Berlin Heidelberg, 2005.
[11] J. Gorski, S. Ruzika, On k-max-optimization, Oper. Res. Lett. 37 (1) (2009) 23–26, http://dx.doi.org/10.1016/j.orl.2008.09.007.
[12] S. Hakimi, Optimum locations of switching centers and the absolute centers and medians of a graph, Oper. Res. 12 (3) (1964) 450–459, http://dx.doi.org/10.1287/opre.12.3.450.
[13] K. Hogan, C. ReVelle, Concepts and applications of backup coverage, Manage. Sci. 32 (11) (1986) 1434–1444, http://dx.doi.org/10.1287/mnsc.32.11.1434.
[14] J. Kalcsics, The multi-facility median problem with Pos/Neg weights on general graphs, Comput. Oper. Res. 38 (3) (2011) 674–682, http://dx.doi.org/10.1016/j.cor.2010.08.002.
[15] J. Kalcsics, S. Nickel, M. Pozo, J. Puerto, A. Rodríguez-Chía, The multicriteria p-facility median location problem on networks, European J. Oper. Res. 235 (3) (2014) 484–493, http://dx.doi.org/10.1016/j.ejor.2014.01.003.
[16] J. Kalcsics, S. Nickel, J. Puerto, Multifacility ordered median problems on networks: A further analysis, Networks 41 (1) (2003) 1–12.
[17] J. Kalcsics, S. Nickel, J. Puerto, A. Rodríguez-Chía, Several 2-facility location problems on networks with equity objectives, Networks 65 (1) (2015) 1–9, http://dx.doi.org/10.1002/net.21568.
[18] O. Kariv, S. Hakimi, An algorithmic approach to network location problems. I: The p-centers, SIAM J. Appl. Math. 37 (3) (1979) 513–538, http://dx.doi.org/10.1137/0137040.
[19] E. Minieka, The m-center problem, SIAM Rev. 12 (1) (1970) 138–139, http://dx.doi.org/10.1137/1012016.
[20] S. Nickel, J. Puerto, A unified approach to network location problems, Networks 34 (4) (1999) 283–290, http://dx.doi.org/10.1002/(SICI)1097-0037(199912)34:4<283::AID-NET8>3.0.CO;2-2.
[21] S. Nickel, J. Puerto, Location Theory: A Unified Approach, Springer Berlin Heidelberg, 2005.
[22] A. Punnen, Y. Aneja, Lexicographic balanced optimization problems, Oper. Res. Lett. 32 (1) (2004) 27–30, http://dx.doi.org/10.1016/0377-2217(91)90073-5.
[23] T. Schnepper, Location Problems with k-max Functions – Modelling and Analysing Outliers in Center Problems (Ph.D. thesis), University of Wuppertal, 2017.
[24] T. Schnepper, K. Klamroth, M. Stiglmayr, J. Puerto, Exact algorithms for handling outliers in center location problems on networks using k-max functions, European J. Oper. Res. 273 (2) (2019) 441–451, http://dx.doi.org/10.1016/j.ejor.2018.08.030, URL: http://www.sciencedirect.com/science/article/pii/S0377221718307252.
[25] P. Sokkalingam, Y. Aneja, Lexicographic bottleneck combinatorial problems, Oper. Res. Lett. 23 (1–2) (1998) 27–33, http://dx.doi.org/10.1016/s0167-6377(98)00028-5.
[26] J.-F. Tsai, M.-H. Lin, Y.-C. Hu, Finding multiple solutions to general integer linear programs, European J. Oper. Res. 184 (2) (2008) 802–809, http://dx.doi.org/10.1016/j.ejor.2006.11.024, URL: http://www.sciencedirect.com/science/article/pii/S0377221706011611.
[27] L. Turner, Variants of shortest path problems, Algorithmic Oper. Res. 6 (2) (2012) 91–104, URL: https://journals.lib.unb.ca/index.php/AOR/article/view/18312.
[28] F. Wang, D. Xu, C. Wu, Combinatorial approximation algorithms for the robust facility location problem with penalties, J. Global Optim. 64 (3) (2016) 483–496, http://dx.doi.org/10.1007/s10898-014-0251-6.
[29] Y. Yamamoto, Optimization over the efficient set: overview, J. Global Optim. 22 (1) (2002) 285–317, http://dx.doi.org/10.1023/A:1013875600711.
[30] H. Zarrabi-Zadeh, A. Mukhopadhyay, Streaming 1-center with outliers in high dimensions, in: Proceedings of the 21st Canadian Conference on Computational Geometry (CCCG 2009), 2009, pp. 83–86.