The Ordered k-Median Problem: Surrogate Models and Approximation Algorithms

Ali Aouad* Danny Segev†

Abstract

Motivated by applications in supply chain, network management, and machine learning, a steady stream of research has been devoted to studying various computational aspects of the ordered k-median problem, which subsumes traditional facility location problems (such as median, center, p-centrum, etc.) through a unified modeling approach. Given a finite metric space, the objective is to locate k facilities in order to minimize the ordered median cost function. In its general form, this function penalizes the coverage distance of each vertex by a multiplicative weight, depending on its ranking (or percentile) in the ordered list of all coverage distances. While antecedent literature has focused on mathematical properties of ordered median functions, integer programming methods, various heuristics, and special cases, this problem was not studied thus far through the lens of approximation algorithms. In particular, even on simple network topologies, such as trees or line graphs, obtaining non-trivial approximation guarantees is an open question.

The main contribution of this paper is to devise the first provably-good approximation algorithms for the ordered k-median problem. We develop a novel approach that relies primarily on a surrogate model, where the ordered median function is replaced by a simplified ranking-invariant functional form, via efficient enumeration. Surprisingly, while this surrogate model is $\Omega(n^{\Omega(1)})$-hard to approximate on general metrics, we obtain an $O(\log n)$-approximation for our original problem by employing local search methods on a smooth variant of the surrogate function. In addition, an improved guarantee of $2+\epsilon$ is obtained on tree metrics by optimally solving the surrogate model through dynamic programming. Finally, we show that the latter optimality gap is tight up to an $O(\epsilon)$ term.

Keywords: Location theory, approximation algorithms, surrogate model, local search.

*London Business School, United Kingdom. Email: aaouad@london.edu

†Department of Statistics, University of Haifa, Haifa 31905, Israel, Email: segevd@stat.haifa.ac.il.

1 Introduction

Motivated by applications in supply chain, network management, and machine learning [36, 35, 8, 38], the ordered k-median problem has been developing in the last two decades as a robust and unified approach to facility location. In this setting, we are given a finite metric space (V, d) on n vertices, a sequence of penalty weights $\lambda_1 \geq \cdots \geq \lambda_n \geq 0$, and an integer parameter k. For any set of facilities $\mathcal{F} \subseteq V$, the ordered median cost function penalizes the distance $d(v, \mathcal{F})$ of each vertex $v \in V$ to its nearest facility by a multiplicative weight, depending on its ranking (or percentile) in the ordered list of all coverage distances. To formalize this criterion, for every ranking $i \in [n]$, let $\Delta_{\mathcal{F}}(i)$ be the i-th largest distance out of $\{d(v, \mathcal{F}) : v \in V\}$, breaking ties arbitrarily. Then, the largest distance $\Delta_{\mathcal{F}}(1)$ is penalized by λ_1, the second largest distance $\Delta_{\mathcal{F}}(2)$ is penalized by λ_2, so forth and so on, ensuring that higher-ranked vertices are penalized more heavily than lower-ranked ones. Consequently, the objective is to compute a set $\mathcal{F} \subseteq V$ of at most k facilities that minimizes the ordered median cost function:

$$\psi(\mathcal{F}) = \sum_{i \in [n]} \lambda_i \cdot \Delta_{\mathcal{F}}(i) \ .$$

Relation to basic models. The ordered median framework develops a uniform and standardized approach to a wide array of facility location problems, where the cost function is expressed in general form, applying arbitrary penalty weights depending on the percentile rank of the coverage distances. This modeling framework subsumes some of the most fundamental objective functions in location theory [11, 23, 3, 29, 9, 22, 48, 49], such as:

- k-median, which minimizes the average distance to the nearest facility, captured by setting $(\lambda_1, \ldots, \lambda_n) = (1, \ldots, 1)$.
- k-center, which minimizes the maximal coverage distance, with $(\lambda_1, \ldots, \lambda_n) = (1, 0, \ldots, 0)$.
- k-centdian, which generalizes convex combinations of k-median and k-center.
- k-facility p-centrum, which minimizes the sum of the p-largest coverage distances, with $\lambda_1 = \cdots = \lambda_p = 1$ and $\lambda_{p+1} = \cdots = \lambda_n = 0$.

In its utmost generality, the ordered median function allows one to formulate a continuum of trade-offs between the k-median and k-center models (see, for example, [37, Chap. 1]), where outlier vertices are penalized according to their percentile rank. For an in-depth discussion of the theory and applications of facility location models, we refer the reader to books on this general topic [34, 13] as well as to those dedicated specifically to ordered median models [27, 37].

Computational challenges. The modeling power discussed above poses additional technical obstacles, in comparison to specialized objectives such as k-median and k-center, which are known to admit constant-factor approximations [11, 23, 3, 29, 9, 22]. The first challenge stems from the dependency of the ordered median cost function on the relative distance rankings. For example, due to the lack of clear separability properties [45], even on tree metrics or line graphs

the ordered k-median problem is not known to admit efficient dynamic programming formulations, unlike k-median and k-center, where such formulations are straightforward. Another inherent challenge is presented by the nonlinearity of the ordered median cost function. In particular, we are not aware of any way to employ probabilistic embeddings into trees [4, 5, 18, 33] to obtain approximation guarantees on general metrics, as further explained in Section 2.

Directly-related work. For these reasons, one line of research has focused on continuous ordered median problems [42, 14, 17, 46, 37], where the metric space is induced by a network, and facilities can be located in the interior of each edge. In such settings, polynomial-time algorithms were obtained for locating a single facility [36, 25, 15, 17], while discrete multi-facility problems were shown to be efficiently solvable under additional structure, such as that combining tree metrics and specific forms of penalty weights [24, 48, 44]. A concurrent line of research has focused on compact integer programming formulations [35], branch-and-bound methods [6, 41], and various heuristics [12, 39, 47]. Along the way, several closely-related models were studied, including line location, hub location, and inverse optimization problems [45, 47, 43, 20, 32, 28, 40, 41]. The book of Nickel and Puerto [37], dedicated to the ordered median problem, provides a detailed review of this literature. Nevertheless, in spite of this substantial body of work, obtaining any non-trivial approximation for the ordered k-median problem is still an intriguing open question, even on simple network topologies, such as trees or line graphs.

Our results. The main contribution of this paper is to devise the first provably-good approximation algorithms for the ordered k-median problem. As further explained in Section 2, we develop a novel approach, that relies primarily on a surrogate model, where the ordered median cost function is replaced by a ranking-invariant functional form. Given an error parameter $\epsilon > 0$, the latter function is constructed through efficient enumeration, while ensuring that the optimality loss resulting from this reduction is bounded by a factor of $2 + \epsilon$.

Surprisingly, while this surrogate model is $\Omega(n^{\Omega(1)})$-hard to approximate by itself, we devise an $O(\log n)$-approximation for our original problem on general metrics. Specifically, we construct a smooth variant of the surrogate function, which is optimized by employing local search methods. In addition, we show that an improved guarantee of $2 + \epsilon$ can be obtained on tree metrics, by optimally solving the surrogate model through dynamic programming. We argue that this optimality gap is tight up to an $O(\epsilon)$ term by introducing a family of instances that fool our surrogate construction into having the desired gap.

2 Technical Overview

As previously mentioned, the main complicating feature of the ordered k-median problem stems from the cost function ψ, which depends on: (1) The distance of each vertex to its nearest facility, and (2) The relative ranking of these vertices with respect to their associated distances. The crux of our method is to construct an alternative cost function which is invariant to the relative distance ranking and only depends on item (1). In this setting, at the expense of incurring a constant-factor loss in optimality, the new cost function can be optimized and analyzed with greater ease, as the cost terms are separable conditional on the individual distance of each

vertex. In what follows, we provide a high-level outline of our algorithmic approach and its analysis.

Step 1: Introducing the surrogate cost function. In Sections 3.1 and 3.2, we introduce an alternative cost function ψ_{sg}, referred to as the surrogate of ψ. To this end, we develop a polynomial-time procedure for guessing certain structural properties of the optimal set of facilities $\mathcal{F}^*$ with respect to the cost function ψ. This structural information is leveraged to eliminate ranking-based dependencies between different vertices, and thus motivates studying the surrogate cost function ψ_{sg}. In order to better fit our approach to various metrics, we also introduce a family of scaled surrogate functions, denoted by $\{\psi_{\text{sg},\alpha}\}$, where $\alpha \geq 1$ is a scaling parameter whose value will be determined according to the type of metric considered.

Step 2: Analyzing the surrogate problem. In Section 3.3, our objective is to bound the optimality gap due to considering the scaled surrogate function $\psi_{\text{sg},\alpha}$, showing that the latter approximates the original cost function ψ within a constant factor. However, as shown in Appendix A, the resulting surrogate problem is $\Omega(n^{\Omega(1)})$-hard to approximate on general metrics. Motivated by this finding, we prove that the desired performance guarantees can alternatively be derived, as long as 'good' solutions for $\psi_{\text{sg},\alpha}$ can be computed efficiently, noting that such 'good' solutions are not approximations in the standard sense. Instead, we utilize the notion of a β-comparable solution, that describes a set of facilities whose surrogate cost is at most a β-factor away from the optimal cost of the original instance (rather than from the optimal surrogate cost).

Step 3a: Approximation on general metrics. On general metrics, we prove that an appropriately scaled surrogate function admits $O(\log n)$-comparable solutions in polynomial time. Technically speaking, such solutions are computed through local search methods, by considering a smooth variant of the scaled surrogate function, created through exponential interpolation. The specifics of our algorithm are given in Section 4.

Theorem 2.1. *On general metrics, the ordered k-median problem can be approximated within factor $O(\log n)$ in polynomial time.*

Step 3b: Approximation on tree metrics. When the underlying metric is induced by a tree, we show that the surrogate function ψ_{sg} can indeed be solved to optimality in polynomial time. For this purpose, we exploit the separable nature of ψ_{sg} to compute an optimal set of facilities by means of dynamic programming. Combined with the established gap between ψ_{sg} and ψ, we derive the next result.

Theorem 2.2. *On tree metrics, the ordered k-median problem can be approximated within factor $2+\epsilon$ in time $O(n^{O((1/\epsilon)\cdot\log(1/\epsilon))})$, for any $\epsilon \in (0,1)$.*

To complement this result, we show that our analysis is essentially tight for tree metrics, by presenting a construction where the optimal surrogate solution has an optimality gap of $2-O(\epsilon)$ with respect to the ordered median cost function. These results are presented in Section 5.

Relationships between tree metrics and general metrics. It is worth noting that we are not aware of any way to derive an approximation guarantee on general metrics using probabilistic embeddings into tree metrics [4, 5, 18]. Indeed, the distortion bounds in such embeddings hold for each single edge, in expectation. However, in our setting, the ordered median cost function is highly non-linear due to its dependency on the relative distance rankings. As a result, evaluating the joint distortion effects across all edges is instrumental to obtain approximation guarantees in this context, which does not seem possible using existing methods. For similar reasons, even stronger distortion guarantees, such as maximum-gradient embeddings [33] or those into spanning trees [2, 16, 1], seem insufficient for our purposes.

3 Surrogate Functions

3.1 Preprocessing

As a preliminary step, we describe a rather standard transformation of the original instance, in order to reduce the variability in the penalty weights $\lambda_1, \ldots, \lambda_n$, at the cost of a negligible loss in optimality. Due to this alteration, the optimal set of facilities could very well change in the modified instance; throughout the paper, we keep using $\mathcal{F}^*$ to denote the optimal set of facilities for the original instance.

Initial guesses. In what follows, we assume that the precise value of $\Delta_{\mathcal{F}^*}(1)$, i.e., the maximal distance between a vertex in V and its nearest facility in $\mathcal{F}^*$, is known in advance. This assumption can be enforced by observing that $\Delta_{\mathcal{F}^*}(1)$ corresponds to the distance between some pair of vertices, and therefore, there are only $O(n^2)$ values to be tested. In addition, given an accuracy parameter $\epsilon \in (0, 1)$, let $i_{\min}$ be the minimal ranking $i \in [n]$ for which $\lambda_i \leq \frac{\epsilon \cdot \lambda_1}{n}$ or $\Delta_{\mathcal{F}^*}(i) \leq \frac{\epsilon \Delta_{\mathcal{F}^*}(1)}{n}$. If none of the rankings satisfies this property, we set $i_{\min} = n + 1$. Note that $i_{\min}$ can easily be guessed by testing each of the values $1, \ldots, n + 1$ as a candidate.

Rounding the penalty weights. We now create a modified sequence of weights $\tilde{\lambda}_1 \geq \cdots \geq \tilde{\lambda}_n$ that dominates $\lambda_1 \geq \cdots \geq \lambda_n$, i.e., $\tilde{\lambda}_i \geq \lambda_i$ for every $i \in [n]$. To this end, the modified weights are obtained by rounding up every λ_i as follows:

$$\tilde{\lambda}_i = \begin{cases} \lambda_i, & \text{if } i < i_{\min} \\ \max\{\lambda_{i_{\min}}, \frac{\epsilon \lambda_1}{n}\}, & \text{if } i \geq i_{\min} \end{cases}$$

Recalling that ψ stands for the cost function with respect to the original weights λ_i, let us denote by $\tilde{\psi}$ the one with respect to the rounded-up weights $\tilde{\lambda}_i$. Given the above rounding operation, $\psi(\mathcal{F}) \leq \tilde{\psi}(\mathcal{F})$ for any set of facilities $\mathcal{F} \subseteq V$. The next lemma, whose proof appears in Appendix B.1, shows that the cost of $\mathcal{F}^*$ under the modified function $\tilde{\psi}$ is at most a $(1 + \epsilon)$-factor away from the original optimal cost.

Lemma 3.1. $\tilde{\psi}(\mathcal{F}^*) \leq (1 + \epsilon) \cdot \psi(\mathcal{F}^*)$.

It follows that any β-approximate solution for $\tilde{\psi}$ provides a $(1 + \epsilon) \cdot \beta$-approximate solution with respect to the original cost function ψ. Therefore, following this modification, we overload

on notation and use $\lambda_1, \ldots, \lambda_n$ to denote the modified weights, in place of $\tilde{\lambda}_1, \ldots, \tilde{\lambda}_n$, and ψ as a substitute to $\tilde{\psi}$. By construction, we have just ensured that $\lambda_{i_{\min}} = \lambda_{i_{\min}+1} = \cdots = \lambda_n$ and $\lambda_n \geq \epsilon\lambda_1/n$.

3.2 Defining the surrogate cost function

To better understand the next steps of our construction, we advise the reader to consult Figure 1.

Partition into distance classes. Recall that the precise value of $\Delta_{\mathcal{F}^*}(1)$ was guessed exactly, as part of the initial preprocessing step. With this parameter at hand, letting $P = \lceil \log_{1+\epsilon}(\frac{n}{\epsilon}) \rceil = O(\frac{1}{\epsilon} \log \frac{n}{\epsilon})$, we partition the interval $(0, \infty)$ geometrically by powers of $1+\epsilon$ into the segments $\{\mathcal{D}_p\}_{p \in \mathbb{Z}}$, where

$$\mathcal{D}_p = \left(\frac{\epsilon \cdot \Delta_{\mathcal{F}^*}(1)}{n} \cdot (1+\epsilon)^{P-p-1}, \frac{\epsilon \cdot \Delta_{\mathcal{F}^*}(1)}{n} \cdot (1+\epsilon)^{P-p} \right] .$$

Note that, by definition of P, the segment $\mathcal{D}_0$ contains the maximal distance $\Delta_{\mathcal{F}^*}(1)$. In the sequel, for any $d \in \mathcal{D}_p$, we say that p is the distance class of d, and use $[p]_0$ to denote the set $\{0, 1, \ldots, p\}$.

Next, for every $p \in [P-1]_0$, we define the collection of rankings $\mathcal{I}_p^* \subseteq [n]$ whose distance value in $\mathcal{F}^*$ resides within the segment $\mathcal{D}_p$, i.e., $\mathcal{I}_p^* = \{i \in [n] : \Delta_{\mathcal{F}^*}(i) \in \mathcal{D}_p\}$. In addition, we define $\mathcal{I}_P^* = \{i \in [n] : \Delta_{\mathcal{F}^*}(i) \in [0, \max \mathcal{D}_P]\}$. Consequently, $\mathcal{I}_0^*, \ldots, \mathcal{I}_P^*$ is a partition of $[n]$ into pairwise-disjoint subintervals in left-to-right order.

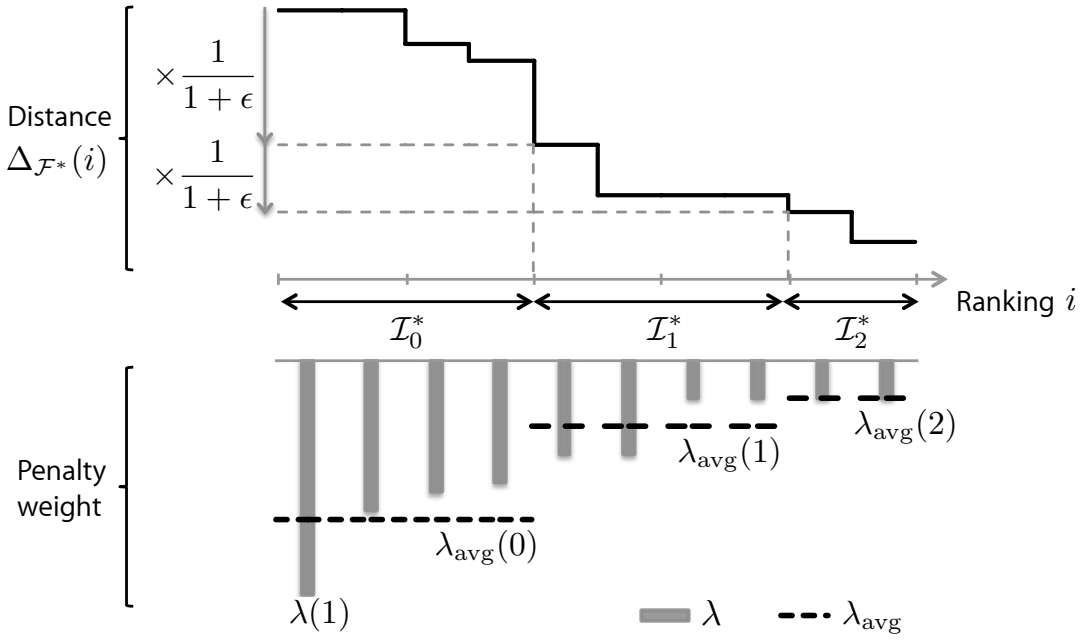

Figure 1: Schematic illustration of how the surrogate function is constructed.

Approximately guessing λ_{avg} values. For every $p \in [P]_0$, we introduce the average penalty weight:

$$\lambda_{\text{avg}}(p) = \begin{cases} \frac{1}{|\mathcal{I}_p^*|} \cdot \sum_{i \in \mathcal{I}_p^*} \lambda_i, & \text{if } \mathcal{I}_p^* \neq \emptyset \\ \min\{\lambda_j : j \in \bigcup_{p' < p} \mathcal{I}_{p'}^*\}, & \text{if } \mathcal{I}_p^* = \emptyset \end{cases} \tag{1}$$

In other words, when the interval $\mathcal{I}_p^*$ is not empty, $\lambda_{\text{avg}}(p)$ is the average penalty weight across $\mathcal{I}_p^*$. Otherwise, this value corresponds to the minimal penalty weight in the first non-empty interval to the left of $\mathcal{I}_p^*$. The latter case is indeed well-defined since the first interval $\mathcal{I}_0^*$ is non-empty, due to having $\Delta_{\mathcal{F}^*}(1) \in \mathcal{D}_0$, as mentioned above.

We now argue that, by considering a polynomial number of possible guesses, we can obtain an over-estimate $\lambda_{\text{guess}}(p)$ for the quantity $\lambda_{\text{avg}}(p)$, simultaneously for all $p \in [P]_0$. As subsequently shown, these estimates will satisfy

$$\frac{1}{1+\epsilon} \cdot \lambda_{\text{guess}}(p) \leq \lambda_{\text{avg}}(p) \leq \lambda_{\text{guess}}(p) \ , \tag{2}$$

and will also form a non-increasing sequence $\lambda_{\text{guess}}(0) \geq \cdots \geq \lambda_{\text{guess}}(P)$. To this end, by definition of λ_{avg}, it is easy to verify that $\lambda_{\text{avg}}(0) \geq \cdots \geq \lambda_{\text{avg}}(P)$. Thus, there exists a non-increasing sequence of integers $\mu_0 \geq \cdots \geq \mu_P$ for which $(1+\epsilon)^{\mu_p - 1} \leq \lambda_{\text{avg}}(p) \leq (1+\epsilon)^{\mu_p}$, where each $(1+\epsilon)^{\mu_p}$ will serve as the value of our estimate $\lambda_{\text{guess}}(p)$. The important observation is that $\mu_0 - \mu_P = O(\frac{1}{\epsilon} \log \frac{n}{\epsilon})$ since $\lambda_1/\lambda_n \leq n/\epsilon$, following the preprocessing step of Section 3.1. Therefore, basic counting arguments imply that the number of sequences $\mu_0 \geq \cdots \geq \mu_P$ to consider is $O(\exp(O((1/\epsilon) \cdot \log(n/\epsilon)))) = O(n^{O((1/\epsilon) \cdot \log(1/\epsilon))})$.

Defining the surrogate cost functions. We begin by defining the penalty function $\lambda_{\text{sg}} : (0, \infty) \to \mathbb{R}^+$ as follows:

$$\lambda_{\text{sg}}(d) = \begin{cases} \lambda_{\text{guess}}(P), & \text{if } d \leq \max \mathcal{D}_P \\ \lambda_{\text{guess}}(p), & \text{if } d \in \mathcal{D}_p \text{ with } p \in [P-1]_0 \\ \lambda_1, & \text{if } d > \max \mathcal{D}_0 \end{cases}$$

Observation 3.2. *λ_{sg} is a non-decreasing left-continuous step function.*

We are now ready to specify the surrogate cost function, for any set of facilities $\mathcal{F} \subseteq V$,

$$\psi_{\text{sg}}(\mathcal{F}) = \sum_{v \in V} \lambda_{\text{sg}}\left(d\left(v, \mathcal{F}\right)\right) \cdot d(v, \mathcal{F}) \ ,$$

where the marginal cost associated with each vertex $v \in V$ is expressed as $\lambda_{\text{sg}}(d(v, \mathcal{F})) \cdot d(v, \mathcal{F})$, which is clearly a function of the distance $d(v, \mathcal{F})$ and nothing more. In addition, we define the scaled penalty function $\lambda_{\text{sg},\alpha}(d) = \lambda_{\text{sg}}(d/\alpha)$, where $\alpha \geq 1$ is the scaling parameter, whose precise value will be determined based on the metric considered, potentially depending on the number of vertices n. Consequently, for every set of facilities $\mathcal{F} \subseteq V$, the scaled surrogate cost function is defined as

$$\psi_{\text{sg},\alpha}(\mathcal{F}) = \sum_{v \in V} \lambda_{\text{sg},\alpha}\left(d\left(v, \mathcal{F}\right)\right) \cdot d(v, \mathcal{F}) = \sum_{i \in [n]} \lambda_{\text{sg},\alpha}\left(\Delta_{\mathcal{F}}\left(i\right)\right) \cdot \Delta_{\mathcal{F}}\left(i\right) \ .$$

The next lemma shows that, regardless of the scaling parameter α, the surrogate cost associated with the optimal set of facilities $\mathcal{F}^*$ is at most a $(1+5\epsilon)$-factor away from the true optimal cost.

Lemma 3.3. *$\psi_{\mathrm{sg},\alpha}(\mathcal{F}^*) \leq (1+5\epsilon)\cdot\psi(\mathcal{F}^*)$, for any $\alpha \geq 1$.*

Proof. Based on the preceding discussion, we have

$$\begin{aligned}
\psi_{\mathrm{sg},\alpha}(\mathcal{F}^*) &= \sum_{i\in[n]} \lambda_{\mathrm{sg}}\left(\frac{\Delta_{\mathcal{F}^*}(i)}{\alpha}\right)\cdot\Delta_{\mathcal{F}^*}(i) \\
&\leq \sum_{i\in[n]} \lambda_{\mathrm{sg}}\left(\Delta_{\mathcal{F}^*}(i)\right)\cdot\Delta_{\mathcal{F}^*}(i) && (3)\\
&= \sum_{p=0}^{P}\sum_{i\in\mathcal{I}_p^*} \lambda_{\mathrm{sg}}\left(\Delta_{\mathcal{F}^*}(i)\right)\cdot\Delta_{\mathcal{F}^*}(i) && (4)\\
&\leq \sum_{p=0}^{P}\sum_{i\in\mathcal{I}_p^*} \lambda_{\mathrm{guess}}(p)\cdot\max\mathcal{D}_p && (5)\\
&\leq (1+\epsilon)\cdot\sum_{p=0}^{P}\left|\mathcal{I}_p^*\right|\cdot\lambda_{\mathrm{avg}}(p)\cdot\max\mathcal{D}_p && (6)\\
&= (1+\epsilon)\cdot\sum_{p=0}^{P}\sum_{i\in\mathcal{I}_p^*}\lambda_i\cdot\max\mathcal{D}_p && (7)\\
&\leq (1+\epsilon)^2\cdot\left(\sum_{p=0}^{P-1}\sum_{i\in\mathcal{I}_p^*}\lambda_i\cdot\Delta_{\mathcal{F}^*}(i)\right) + (1+\epsilon)\cdot|\mathcal{I}_P^*|\cdot\lambda_1\cdot\frac{\epsilon\Delta_{\mathcal{F}^*}(1)}{n} && (8)\\
&\leq (1+5\epsilon)\cdot\psi(\mathcal{F}^*)\ .
\end{aligned}$$

Here, inequality (3) holds since $\alpha \geq 1$ and since the function λ_{sg} is non-decreasing (Observation 3.2). Equality (4) holds since the intervals $\mathcal{I}_0^*,\ldots,\mathcal{I}_P^*$ form a partition of $[n]$. The next inequality (5) follow from the definition of λ_{sg}, while inequality (6) holds due to the accuracy of our guessing procedure (2). Equality (7) is obtained by observing that, for non-empty intervals $\mathcal{I}_p^*$, we have $|\mathcal{I}_p^*|\cdot\lambda_{\mathrm{avg}}(p) = \sum_{i\in\mathcal{I}_p^*}\lambda_i$ (see (1)). Inequality (8) holds since, when $p\in[P-1]_0$, each of the rankings $i\in\mathcal{I}_p^*$ has $\Delta_{\mathcal{F}^*}(i)\in\mathcal{D}_p$, and therefore, $\max\mathcal{D}_p \leq (1+\epsilon)\cdot\Delta_{\mathcal{F}^*}(i)$, whereas for the last interval $\mathcal{I}_P^*$, we have by construction $\max\mathcal{D}_P = \frac{\epsilon}{n}\cdot\Delta_{\mathcal{F}^*}(1)$. ∎

3.3 Relating between the original and surrogate functions

We say that a feasible set of facilities $\mathcal{F}\subseteq V$ is a β-comparable solution for a cost function $\hat{\psi}$ if $\hat{\psi}(\mathcal{F}) \leq \beta\cdot\psi(\mathcal{F}^*)$. In contrast to a standard approximation, the latter inequality compares the surrogate cost $\hat{\psi}(\mathcal{F})$ directly against the original optimal cost $\psi(\mathcal{F}^*)$, which is generally different than the optimal surrogate cost. In the remainder of this section, we argue that any β-comparable solution for the surrogate cost function $\psi_{\mathrm{sg},\alpha}$ provides an $O(\max\{\alpha,\beta\})$-approximation for the original ordered k-median problem, as stated in the next theorem. In particular, this result implies that any optimal solution for the surrogate cost function ψ_{sg} is a $(2+O(\epsilon))$-approximation with respect to the original instance.

Theorem 3.4. *Let $\mathcal{F} \subseteq V$ be a β-comparable set of facilities for the cost function $\psi_{\mathrm{sg},\alpha}$. Then, $\psi(\mathcal{F}) \leq (2+2\epsilon) \cdot \max\{\alpha, \beta\} \cdot \psi(\mathcal{F}^*)$.*

Benchmark sequence. The analysis proceeds by comparing how the sequence of ranked distances $\Delta_{\mathcal{F}}(1), \ldots, \Delta_{\mathcal{F}}(n)$ evolves with respect to an appropriate benchmark, chosen as

$$\Delta^{*\alpha} = (\ \alpha \cdot \Delta_{\mathcal{F}^*}(1)\ ,\ \alpha \cdot \Delta_{\mathcal{F}^*}(2)\ ,\ \ldots\ ,\ \alpha \cdot \Delta_{\mathcal{F}^*}(n)\ )\ .$$

Note that this ranked sequence of distances does not necessarily correspond to any feasible set of facilities; it is only defined to be compared with $\Delta_{\mathcal{F}}$, for purposes of analysis. In addition, we define a cost value associated with the benchmark:

$$\varphi\left(\Delta^{*\alpha}\right) = \sum_{i \in [n]} \lambda_i \cdot (\alpha \cdot \Delta_{\mathcal{F}^*}(i)) = \alpha \cdot \psi\left(\mathcal{F}^*\right)\ . \tag{9}$$

Auxiliary function. We define an auxiliary function $\mathrm{sign} : [n] \to \{+, -, 0\}$, that allows us to identify the rankings where the sequence $\Delta_{\mathcal{F}}(1), \ldots, \Delta_{\mathcal{F}}(n)$ significantly deviates from the benchmark sequence $\Delta^{*\alpha}$:

$$\mathrm{sign}(i) = \begin{cases} +, & \text{if } \Delta_{\mathcal{F}}(i) < \alpha\Delta_{\mathcal{F}^*}(i)/(1+\epsilon)\ , \\ -, & \text{if } \Delta_{\mathcal{F}}(i) > (1+\epsilon) \cdot \alpha\Delta_{\mathcal{F}^*}(i)\ , \\ 0, & \text{if } \alpha\Delta_{\mathcal{F}^*}(i)/(1+\epsilon) \leq \Delta_{\mathcal{F}}(i) \leq (1+\epsilon) \cdot \alpha\Delta_{\mathcal{F}^*}(i)\ . \end{cases}$$

The next lemma establishes certain 'stabilization' properties, which are crucial ingredients of our analysis. Essentially, whenever the sequence $\Delta_{\mathcal{F}}$ is beaten by the benchmark $\Delta^{*\alpha}$, i.e., the sign function is negative, the surrogate cost $\psi_{\mathrm{sg},\alpha}$ applies a penalty weight larger than the true cost ψ. Thus, the surrogate setting is being conservative by overestimating the penalty weights. Conversely, whenever the sequence $\Delta_{\mathcal{F}}$ outperforms $\Delta^{*\alpha}$, i.e., the sign function is positive, $\psi_{\mathrm{sg},\alpha}$ applies a penalty weight smaller than in ψ. Here, the surrogate setting is conservative in a different way, by diminishing the fraction of the cost where $\Delta_{\mathcal{F}}$ is smaller than $\Delta^{*\alpha}$.

Lemma 3.5. *For every ranking $i \in [n]$, we have:*

1. *If* $\mathrm{sign}(i) = -$ *then* $\lambda_{\mathrm{sg},\alpha}(\Delta_{\mathcal{F}}(i)) \geq \lambda_i$.
2. *If* $\mathrm{sign}(i) = +$ *then* $\lambda_{\mathrm{sg},\alpha}(\Delta_{\mathcal{F}}(i)) \leq (1+\epsilon) \cdot \lambda_i$.

Proof. To prove claim 1, consider some $i \in [n]$ for which $\mathrm{sign}(i) = -$. By definition of the sign function, we have $\Delta_{\mathcal{F}}(i) > (1+\epsilon) \cdot \alpha \cdot \Delta_{\mathcal{F}^*}(i)$. In this case,

$$\lambda_{\mathrm{sg},\alpha}\left(\Delta_{\mathcal{F}}(i)\right) = \lambda_{\mathrm{sg}}\left(\frac{\Delta_{\mathcal{F}}(i)}{\alpha}\right) \geq \lambda_{\mathrm{sg}}\left((1+\epsilon) \cdot \Delta_{\mathcal{F}^*}(i)\right)\ ,$$

where the last equality holds due to the monotonicity of λ_{sg} (Observation 3.2). From this point on, we distinguish between three cases based on the value of $\pi^*(i) \in [P]_0$, which denotes the index of the interval in $\mathcal{I}_0^*, \ldots, \mathcal{I}_P^*$ that contains i, i.e., $i \in \mathcal{I}_{\pi^*(i)}^*$.

Case 1: $\pi^*(i) = 0$. Since the distance classes $\{\mathcal{D}_p\}_{p\in\mathbb{Z}}$ are delimited by powers of $1+\epsilon$, it follows that $(1+\epsilon)\cdot\Delta_{\mathcal{F}^*}(i) > \max\mathcal{D}_{\pi^*(i)} = \max\mathcal{D}_0$. Therefore, $\lambda_{\text{sg}}((1+\epsilon)\cdot\Delta_{\mathcal{F}^*}(i)) = \lambda_1$, and we get $\lambda_{\text{sg},\alpha}(\Delta_{\mathcal{F}}(i)) \geq \lambda_1 \geq \lambda_i$, where the last inequality holds since $\lambda_1 \geq \cdots \geq \lambda_n$.

Case 2: $\pi^*(i) = P$. In this case, the distance class of $(1+\epsilon)\cdot\Delta_{\mathcal{F}^*}(i)$ is greater or equal to $P-1$. Hence, we obtain

$$\begin{aligned}
\lambda_{\text{sg}}\left((1+\epsilon)\cdot\Delta_{\mathcal{F}^*}(i)\right) &\geq \min\left\{\lambda_{\text{guess}}(P-1), \lambda_{\text{guess}}(P)\right\} \\
&= \lambda_{\text{guess}}(P) \\
&\geq \lambda_{\text{avg}}(P) \\
&= \frac{1}{|\mathcal{I}_P^*|}\cdot\sum_{j\in\mathcal{I}_P^*}\lambda_j \\
&= \lambda_i \ ,
\end{aligned}$$

where the first inequality holds since $\lambda_{\text{guess}}(0) \geq \cdots \geq \lambda_{\text{guess}}(P)$, and the second inequality is due to (2). The second equality proceeds from the definition of λ_{avg} (see (1)), by observing that the interval $\mathcal{I}_P^*$ is non-empty since in particular $i \in \mathcal{I}_P^*$. To understand the last equality, note that $i \in \mathcal{I}_P^*$ and the penalty weights are uniform across the interval $\mathcal{I}_P^*$, since by our preprocessing step $\lambda_{i_{\min}} = \lambda_{i_{\min}+1} = \cdots = \lambda_n$ and $\mathcal{I}_P^* \subseteq [i_{\min}, n]$. Indeed, for any $j \in \mathcal{I}_P^*$, we have $\Delta_{\mathcal{F}^*}(j) \leq \epsilon\Delta_{\mathcal{F}^*}(1)/n$, meaning that $j \geq i_{\min}$, given the definition of $i_{\min}$ in Section 3.1.

Case 3: $1 \leq \pi^*(i) \leq P-1$. In this case, we necessarily have $\Delta_{\mathcal{F}^*}(i) \in \mathcal{D}_{\pi^*(i)}$, and therefore $(1+\epsilon)\cdot\Delta_{\mathcal{F}^*}(i) \in \mathcal{D}_{\pi^*(i)-1}$. Hence,

$$\lambda_{\text{sg}}\left((1+\epsilon)\cdot\Delta_{\mathcal{F}^*}(i)\right) = \lambda_{\text{guess}}\left(\pi^*(i)-1\right) \geq \lambda_{\text{avg}}\left(\pi^*(i)-1\right) \geq \lambda_i \ ,$$

where the first inequality holds due to the accuracy of the guessing procedure (2). To understand the last inequality, observe that: (i) when $\mathcal{I}^*_{\pi^*(i)-1}$ is non-empty, $\lambda_{\text{avg}}\left(\pi^*(i)-1\right)$ is defined as the average penalty weight in the interval $\mathcal{I}^*_{\pi^*(i)-1}$; (ii) when $\mathcal{I}^*_{\pi^*(i)-1}$ is empty, $\lambda_{\text{avg}}\left(\pi^*(i)-1\right)$ is the minimal penalty weight in the first non-empty interval to the left of $\mathcal{I}^*_{\pi^*(i)}$. The inequality follows from the position of ranking i relative to the intervals considered, since $\lambda_1 \geq \cdots \geq \lambda_n$.

Since the proof of claim 2 makes uses of symmetrical arguments, it is deferred to Appendix B.2. ■

Decomposition of cost functions. We now describe a decomposition of the cost functions defined so far, by slicing the sequence of distances according to the sign function value:

$$\begin{aligned}
\psi_{\text{sg},\alpha}(\mathcal{F}) \ = \ & \underbrace{\sum_{i:\text{sign}(i)=+}\lambda_{\text{sg},\alpha}\left(\Delta_{\mathcal{F}}(i)\right)\cdot\Delta_{\mathcal{F}}(i)}_{\psi^+_{\text{sg},\alpha}(\mathcal{F})} + \underbrace{\sum_{i:\text{sign}(i)=-}\lambda_{\text{sg},\alpha}\left(\Delta_{\mathcal{F}}(i)\right)\cdot\Delta_{\mathcal{F}}(i)}_{\psi^-_{\text{sg},\alpha}(\mathcal{F})} \\
& + \underbrace{\sum_{i:\text{sign}(i)=0}\lambda_{\text{sg},\alpha}\left(\Delta_{\mathcal{F}}(i)\right)\cdot\Delta_{\mathcal{F}}(i)}_{\psi^0_{\text{sg},\alpha}(\mathcal{F})} \ ,
\end{aligned}$$

$$\psi(\mathcal{F}) = \underbrace{\sum_{i:\text{sign}(i)=+} \lambda_i \cdot \Delta_{\mathcal{F}}(i)}_{\psi^+(\mathcal{F})} + \underbrace{\sum_{i:\text{sign}(i)=-} \lambda_i \cdot \Delta_{\mathcal{F}}(i)}_{\psi^-(\mathcal{F})} + \underbrace{\sum_{i:\text{sign}(i)=0} \lambda_i \cdot \Delta_{\mathcal{F}}(i)}_{\psi^0(\mathcal{F})} ,$$

$$\varphi(\Delta^{*\alpha}) = \underbrace{\sum_{i:\text{sign}(i)=+} \lambda_i \cdot \alpha \cdot \Delta_{\mathcal{F}^*}(i)}_{\varphi^+(\Delta^{*\alpha})} + \underbrace{\sum_{i:\text{sign}(i)=-} \lambda_i \cdot \alpha \cdot \Delta_{\mathcal{F}^*}(i)}_{\varphi^-(\Delta^{*\alpha})} + \underbrace{\sum_{i:\text{sign}(i)=0} \lambda_i \cdot \alpha \cdot \Delta_{\mathcal{F}^*}(i)}_{\varphi^0(\Delta^{*\alpha})} .$$

With these definitions at hand, we highlight a number of cost comparisons on the different slices, which are straightforward implications of Lemma 3.5, combined with the definition of the sign function.

Observation 3.6. $\varphi^0(\Delta^{*\alpha})/(1+\epsilon) \leq \psi^0(\mathcal{F}) \leq (1+\epsilon) \cdot \varphi^0(\Delta^{*\alpha})$.

Observation 3.7. $\psi^+(\mathcal{F}) \leq \varphi^+(\Delta^{*\alpha})/(1+\epsilon)$ *and* $\psi^-(\mathcal{F}) \geq (1+\epsilon) \cdot \varphi^-(\Delta^{*\alpha})$.

Observation 3.8. $\psi^-(\mathcal{F}) \leq \psi^-_{\text{sg},\alpha}(\mathcal{F})$ *and* $(1+\epsilon) \cdot \psi^+(\mathcal{F}) \geq \psi^+_{\text{sg},\alpha}(\mathcal{F})$.

Concluding the analysis. We can now complete the proof of Theorem 3.4, by showing that $\psi(\mathcal{F}) \leq (2+2\epsilon) \cdot \max\{\alpha, \beta\} \cdot \psi(\mathcal{F}^*)$. For this purpose, we examine two cases:

1. $\boldsymbol{\psi^+(\mathcal{F}) + (1+\epsilon) \cdot \varphi^0(\Delta^{*\alpha}) \geq \psi^-_{\text{sg},\alpha}(\mathcal{F})}$**.** Here,

$$\varphi^+(\Delta^{*\alpha}) \geq \psi^+(\mathcal{F}) \geq \psi^-_{\text{sg},\alpha}(\mathcal{F}) - (1+\epsilon) \cdot \varphi^0(\Delta^{*\alpha}) , \tag{10}$$

   where the first inequality is due to Observation 3.7, and the second inequality holds given the case hypothesis. Consequently,

$$\begin{aligned} \psi(\mathcal{F}) &= \psi^+(\mathcal{F}) + \psi^-(\mathcal{F}) + \psi^0(\mathcal{F}) \\ &\leq \varphi^+(\Delta^{*\alpha}) + \psi^-_{\text{sg},\alpha}(\mathcal{F}) + (1+\epsilon) \cdot \varphi^0(\Delta^{*\alpha}) \\ &\leq 2 \cdot \varphi^+(\Delta^{*\alpha}) + 2(1+\epsilon) \cdot \varphi^0(\Delta^{*\alpha}) \\ &\leq 2(1+\epsilon) \cdot \varphi(\Delta^{*\alpha}) \\ &= 2(1+\epsilon) \cdot \alpha \cdot \psi(\mathcal{F}^*) . \end{aligned}$$

   The first inequality proceeds from combining Observations 3.6, 3.7, and 3.8, the second inequality follows from (10), and the last equality follows from (9).

2. $\boldsymbol{\psi^+(\mathcal{F}) + (1+\epsilon) \cdot \varphi^0(\Delta^{*\alpha}) < \psi^-_{\text{sg},\alpha}(\mathcal{F})}$**.** In this case, we obtain

$$\begin{aligned} \psi(\mathcal{F}) &= \psi^+(\mathcal{F}) + \psi^-(\mathcal{F}) + \psi^0(\mathcal{F}) \\ &\leq 2 \cdot \psi^-_{\text{sg},\alpha}(\mathcal{F}) - (1+\epsilon) \cdot \varphi^0(\Delta^{*\alpha}) + \psi^0(\mathcal{F}) \\ &\leq 2 \cdot \psi^-_{\text{sg},\alpha}(\mathcal{F}) \\ &\leq 2 \cdot \psi_{\text{sg},\alpha}(\mathcal{F}) \\ &\leq 2\beta \cdot \psi(\mathcal{F}^*) , \end{aligned}$$

where the first inequality holds by the case hypothesis and Observation 3.8, the second inequality is due to Observation 3.6, and the last inequality holds since $\mathcal{F}$ is a β-comparable solution for $\psi_{\mathrm{sg},\alpha}$.

4 General Metrics: Computing $O(\log n)$-Comparable Solutions

In this section, we establish Theorem 2.1 by obtaining an $O(\log n)$-approximation for the ordered k-median problem on general metrics. Our algorithm relies on computing an $O(\log n)$-comparable solution for the surrogate cost function $\psi_{\mathrm{sg},\alpha}$ with logarithmic scaling, through a single-swap local search procedure.

Theorem 4.1. *There is a polynomial-time algorithm to compute an $O(\log n)$-comparable solution for the surrogate cost function $\psi_{\mathrm{sg},\alpha}$ on general metrics, with $\alpha = 80 \log n$.*

By combining the above result with Theorem 3.4, we obtain an $O(\log n)$-approximation for the ordered k-median problem on general metrics. It is worth pointing out that the notion of comparable solutions is indeed necessary, and in fact, we prove in Appendix A that the surrogate model cannot be efficiently approximated (in the standard sense) within factor $O(n^{\Omega(1)})$, unless NP $\subseteq$ TIME$(n^{O(\log \log n)})$.

Definitions and notation. In the remainder of this section, rather than dragging cumbersome expressions throughout the analysis, the error parameter is fixed to $\epsilon = 1$. In addition, we define the segment $\mathcal{D}_p^\alpha = \{x \in (0,\infty) : x/\alpha \in \mathcal{D}_p\}$, corresponding to each of the original segments $\{\mathcal{D}_p\}_{p\in\mathbb{Z}}$. Note that, due to picking $\epsilon = 1$, the segments $\mathcal{D}_p^\alpha$ are now delimited by powers of 2. That is, letting $\delta_p = \max \mathcal{D}_p^\alpha$, we have $\delta_{p-1}/\delta_p = 1 + \epsilon = 2$.

4.1 Smoothing the penalty function

To better understand the upcoming discussion, we advise the reader to consult Figure 2. Due to Observation 3.2, and the definition of the scaled surrogate penalties, $\lambda_{\mathrm{sg},\alpha}$ is a non-decreasing left-continuous step function defined over $(0,\infty)$, which is constant over each distance segment $\mathcal{D}_p^\alpha$. Now, for any distance value $d \in (0,\infty)$, there exists a unique integer p such that $d \in \mathcal{D}_p^\alpha$, as well as a unique real $\gamma \in (1,2]$ such that $d = (\gamma/2)\cdot\delta_p$. With this notation at hand, we define the function $\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}(d) = \lambda_{\mathrm{sg},\alpha}(\delta_p)\cdot\eta_p^{\gamma-1}$, where $\eta_p = \lambda_{\mathrm{sg},\alpha}(\delta_{p-1})/\lambda_{\mathrm{sg},\alpha}(\delta_p)$. Finally, we define the smooth surrogate cost function $\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}$, for every set of facilities $\mathcal{F} \subseteq V$,

$$\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}(\mathcal{F}) = \sum_{v\in V} \lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d(v,\mathcal{F})\right)\cdot d(v,\mathcal{F}) \ .$$

4.2 Properties of the smooth surrogate function

Bounded increment. Note that, due to the preprocessing step in Section 3.1, we necessarily have $\lambda_1/\lambda_n \le n/\epsilon = n$. As a result, by definition of the penalty function $\lambda_{\mathrm{sg},\alpha}$ in Section 3.2, it is easy to verify that the ratio between the extremal values of $\lambda_{\mathrm{sg},\alpha}$ is upper bounded by n as well, meaning in particular that $\eta_p = \lambda_{\mathrm{sg},\alpha}(\delta_{p-1})/\lambda_{\mathrm{sg},\alpha}(\delta_p) \le n$ for any $p \in \mathbb{Z}$. We make use

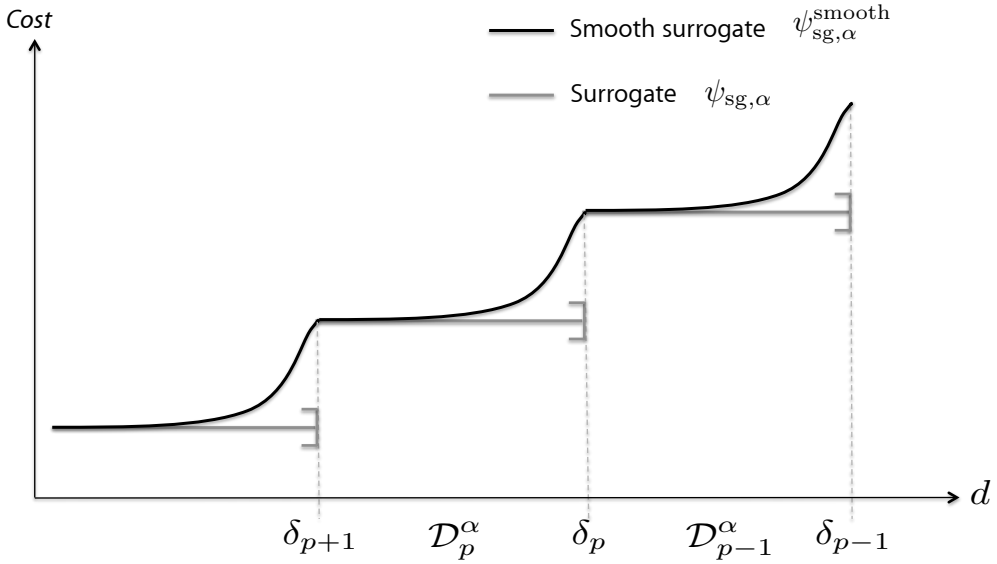

Figure 2: Schematic illustration of $\psi_{\text{sg},\alpha}$ and its smooth approximation $\psi^{\text{smooth}}_{\text{sg},\alpha}$.

of this observation to bound the multiplicative increments of $\lambda^{\text{smooth}}_{\text{sg},\alpha}$ in the next lemma, whose proof is given in Appendix B.3.

Lemma 4.2. $\lambda^{\text{smooth}}_{\text{sg},\alpha}(d')/\lambda^{\text{smooth}}_{\text{sg},\alpha}(d) \leq n^{2(d'/d-1)}$ *for any two distances* $d' \geq d$.

Transfer of β-comparability. We also observe that, for any distance $d \in \mathcal{D}^{\alpha}_{p}$, we have $\lambda_{\text{sg},\alpha}(d) \leq \lambda_{\text{sg},\alpha}(\delta_p) \leq \lambda^{\text{smooth}}_{\text{sg},\alpha}(d)$. Here, the first inequality holds since $\lambda_{\text{sg},\alpha}$ is non-decreasing while $d \leq \max \mathcal{D}^{\alpha}_{p} = \delta_p$, and the second inequality follows from the definition of $\lambda^{\text{smooth}}_{\text{sg},\alpha}$, given that $\eta_p \geq 1$. Therefore, $\lambda^{\text{smooth}}_{\text{sg},\alpha}$ upper bounds $\lambda_{\text{sg},\alpha}$, meaning that any β-comparable solution for $\psi^{\text{smooth}}_{\text{sg},\alpha}$ is β-comparable for $\psi_{\text{sg},\alpha}$ as well.

Additional upper bounds. Finally, we remark that for any distance $d \in \mathcal{D}^{\alpha}_{p}$,

$$\lambda^{\text{smooth}}_{\text{sg},\alpha}(d) \leq \lambda_{\text{sg},\alpha}(\delta_{p-1}) = \lambda_{\text{sg},\alpha}(2d) = \lambda_{\text{sg}}\left(\frac{2d}{\alpha}\right) \leq \lambda_{\text{sg}}(d) \ . \tag{11}$$

Here, the first inequality follows from the definition of $\lambda^{\text{smooth}}_{\text{sg},\alpha}$, the first equality holds since $2d \in \mathcal{D}^{\alpha}_{p-1}$, and the second inequality is due to the scaling factor $\alpha = 80 \log n \geq 2$ and the monotonicity of λ_{sg} (Observation 3.2). In particular, we infer that

$$\begin{aligned}
\psi^{\text{smooth}}_{\text{sg},\alpha}(\mathcal{F}^*) &= \sum_{v \in V} \lambda^{\text{smooth}}_{\text{sg},\alpha}\left(d\left(v, \mathcal{F}^*\right)\right) \cdot d\left(v, \mathcal{F}^*\right) \\
&\leq \sum_{v \in V} \lambda_{\text{sg}}\left(d\left(v, \mathcal{F}^*\right)\right) \cdot d\left(v, \mathcal{F}^*\right) \\
&= \psi_{\text{sg}}(\mathcal{F}^*) \\
&\leq 6 \cdot \psi(\mathcal{F}^*) \ ,
\end{aligned} \tag{12}$$

where the last inequality is due to Lemma 3.3, specialized with $\epsilon = 1$ and $\psi_{\text{sg}} = \psi_{\text{sg},1}$.

4.3 Local search

Algorithm. In order to construct an $O(\log n)$-comparable solution for the smooth surrogate function $\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}$, we make use of the single-swap local search algorithm originally proposed by Charikar and Guha [10] for the uncapacitated facility location problem, and by Arya et al. [3] for the metric k-median problem. By defining a local swap as the operation of closing one facility and opening a new one instead, this algorithm picks in each step a local swap with maximal cost reduction, until the marginal cost change becomes smaller than some pre-defined accuracy level, when a local minimum is nearly reached. For simplicity of presentation, we assume that this local search procedure is executed until all local swaps do not result in cost reduction, and designate by $\hat{\mathcal{F}}$ the set of facilities corresponding to the local minimum reached.

Properties of locally-optimal solutions. We will utilize certain ingredients of the analysis developed by Gupta and Tangwongsan [21], who compared the cost of $\hat{\mathcal{F}}$ against any benchmark solution, by considering an appropriate collection of swaps between the facilities of $\hat{\mathcal{F}}$ and those of the benchmark. In their setting, the objective was to establish a performance guarantee relative to the optimal solution, which was therefore picked as the benchmark. However, in order to prove that $\hat{\mathcal{F}}$ is an $O(\log n)$-comparable solution for $\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}$, here we pick the benchmark as the optimal set of facilities $\mathcal{F}^*$ for the original problem, which could be suboptimal for the current cost function $\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}$.

Specifically, Gupta and Tangwongsan constructed a collection of swaps $(\hat{f}_1, f_1^*), \ldots, (\hat{f}_k, f_k^*)$, where each $\hat{f}_\ell$ is a vertex of $\hat{\mathcal{F}}$ and each f_ℓ^* is a vertex of $\mathcal{F}^*$. For every $\ell \in [k]$, let $\hat{\mathcal{F}}_\ell = (\hat{\mathcal{F}} \setminus \{\hat{f}_\ell\}) \cup \{f_\ell^*\}$ be the set of facilities obtained from $\hat{\mathcal{F}}$ by closing $\hat{f}_\ell$ and opening f_ℓ^*. As explained below, corresponding to each swap $(\hat{f}_\ell, f_\ell^*)$, they also defined a reallocation function $\mathcal{A}_\ell : V \to \hat{\mathcal{F}}_\ell$, that maps each vertex to a 'sufficiently close' facility in $\hat{\mathcal{F}}_\ell$. For any facility $\hat{f} \in \hat{\mathcal{F}}$, let $\hat{N}(\hat{f})$ be the set of vertices in V whose nearest facility in $\hat{\mathcal{F}}$ is $\hat{f}$; similarly, $N^*(f^*)$ stands for the set of vertices in V whose nearest facility in $\mathcal{F}^*$ is f^*. In slight abuse of notation, we make use of $\hat{N}^{-1}(v)$ to denote the facility in $\hat{\mathcal{F}}$ nearest v, while $N^{*-1}(v)$ is defined in an analogous way for $\mathcal{F}^*$. Their construction satisfies the following properties:

1. Each facility $f^* \in \mathcal{F}^*$ occurs exactly once in $f_1^*, \ldots, f_k^*$, while each facility $\hat{f} \in \hat{\mathcal{F}}$ occurs at most twice in $\hat{f}_1, \ldots, \hat{f}_k$.

2. Focusing on a single local swap $(\hat{f}_\ell, f_\ell^*)$, the reallocation function $\mathcal{A}_\ell : V \to \hat{\mathcal{F}}_\ell$ is defined as follows:

   (a) For every vertex $v \in N^*(f_\ell^*)$, we have $\mathcal{A}_\ell(v) = f_\ell^*$.

   (b) For every vertex $v \in \hat{N}(\hat{f}_\ell) \setminus N^*(f_\ell^*)$, we have $\mathcal{A}_\ell(v) = \hat{N}^{-1}(N^{*-1}(v))$, and moreover, this facility is shown to be different from $\hat{f}_\ell$.

   (c) For every vertex $v \in V \setminus (\hat{N}(\hat{f}_\ell) \cup N^*(f_\ell^*))$, we have $\mathcal{A}_\ell(v) = \hat{N}^{-1}(v)$.

When the distance function d is a metric, as in our case, Gupta and Tangwongsan further established the projection lemma [21, Lem. 2.4], which ensures in particular for every $v \in \hat{N}(\hat{f}_\ell) \setminus N^*(f_\ell^*)$ that $d(v, \mathcal{A}_\ell(v)) \le \bar{d}(v)$, where $\bar{d}(v) = d(v, \hat{\mathcal{F}}) + 2 \cdot d(v, \mathcal{F}^*)$. It is important to

note that these structural properties hold regardless of the cost function being optimized, and thus far, the specifics of $\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}$ have not come into play.

Bounding the cost of $\hat{F}_\ell$. The cost variation due to the local swap $(\hat{f}_\ell, f_\ell^*)$ is bounded by

$$\begin{aligned}
&\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(\hat{\mathcal{F}}_\ell\right) - \psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(\hat{\mathcal{F}}\right) \\
&\quad= \sum_{v\in V}\left(\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,\hat{\mathcal{F}}_\ell\right)\right)\cdot d\left(v,\hat{\mathcal{F}}_\ell\right) - \lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,\hat{\mathcal{F}}\right)\right)\cdot d\left(v,\hat{\mathcal{F}}\right)\right) \\
&\quad\le \sum_{v\in V}\left(\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,\mathcal{A}_\ell\left(v\right)\right)\right)\cdot d\left(v,\mathcal{A}_\ell\left(v\right)\right) - \lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,\hat{\mathcal{F}}\right)\right)\cdot d\left(v,\hat{\mathcal{F}}\right)\right) \\
&\quad= \sum_{v\in N^*(f_\ell^*)\cup\hat{N}(\hat{f}_\ell)}\left(\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,\mathcal{A}_\ell\left(v\right)\right)\right)\cdot d\left(v,\mathcal{A}_\ell\left(v\right)\right) - \lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,\hat{\mathcal{F}}\right)\right)\cdot d\left(v,\hat{\mathcal{F}}\right)\right) \\
&\quad\le \sum_{v\in N^*(f_\ell^*)}\left(\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,f_\ell^*\right)\right)\cdot d\left(v,f_\ell^*\right) - \lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,\hat{\mathcal{F}}\right)\right)\cdot d\left(v,\hat{\mathcal{F}}\right)\right) \\
&\quad\quad+ \sum_{v\in\hat{N}(\hat{f}_\ell)}\left(\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(\bar{d}(v)\right)\cdot\bar{d}(v) - \lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(d\left(v,\hat{\mathcal{F}}\right)\right)\cdot d\left(v,\hat{\mathcal{F}}\right)\right) \ . \qquad (13)
\end{aligned}$$

Here, the first inequality is obtained by combining the monotonicity of $\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}$ with $d(v,\hat{\mathcal{F}}_\ell) \le d(v,A_\ell(v))$. The next equality is a consequence of property 2(c). To understand the last inequality, observe that if $v \in N^*(f_\ell^*)$ then $\mathcal{A}_\ell(v) = f_\ell^*$ by property 2(a), and if $v \in \hat{N}(\hat{f}_\ell) \setminus N^*(f_\ell^*)$ then $d(v,\mathcal{A}_\ell(v)) \le \bar{d}(v)$ by combining property 2(b) with the projection lemma.

Now, since $\hat{\mathcal{F}}$ is a local minimum, for each local swap $(\hat{f}_\ell, f_\ell^*)$ there is no cost reduction, i.e., $\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}(\hat{\mathcal{F}}) \le \psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}(\hat{\mathcal{F}}_\ell)$. Therefore, the right-hand side of inequality (13) is non-negative. By summing over the local swaps $(\hat{f}_1, f_1^*), \ldots, (\hat{f}_k, f_k^*)$ and invoking property 1, we finally obtain:

$$3\cdot\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(\hat{\mathcal{F}}\right) - \psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(\mathcal{F}^*\right) \le 2\cdot\sum_{v\in V}\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(\bar{d}\left(v\right)\right)\cdot\bar{d}\left(v\right) \ . \qquad (14)$$

The next claim provides an upper bound on the latter term.

Lemma 4.3. $\sum_{v\in V}\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}(\bar{d}(v))\cdot\bar{d}(v) \le 3\alpha\cdot\psi(\mathcal{F}^*) + 1.27\cdot\psi_{\mathrm{sg},\alpha}^{\mathrm{smooth}}(\hat{\mathcal{F}})$.

Proof. We begin by partitioning the set of vertices V to two subsets $\mathcal{V}_1$ and $\mathcal{V}_2$, where $\mathcal{V}_1$ contains all vertices for which $d(v,\hat{\mathcal{F}}) \le 20\log n\cdot d(v,\mathcal{F}^*)$, and $\mathcal{V}_2$ contains the remaining ones, where $d(v,\hat{\mathcal{F}}) > 20\log n\cdot d(v,\mathcal{F}^*)$.

Bounding the sum over $\mathcal{V}_1$. Observe that for every vertex $v \in \mathcal{V}_1$,

$$\bar{d}(v) \le (20\log n + 2)\cdot d\left(v,\mathcal{F}^*\right) \le 40\log n\cdot d\left(v,\mathcal{F}^*\right) = \frac{\alpha}{2}\cdot d\left(v,\mathcal{F}^*\right) \ , \qquad (15)$$

where the first inequality follows from the definitions of $\bar{d}(v)$ and $\mathcal{V}_1$, while the last equality is obtained by recalling that $\alpha = 80\log n$. As a result, we infer that

$$\lambda_{\mathrm{sg},\alpha}^{\mathrm{smooth}}\left(\bar{d}(v)\right) \le \lambda_{\mathrm{sg},\alpha}\left(2\bar{d}(v)\right) = \lambda_{\mathrm{sg}}\left(\frac{2\bar{d}(v)}{\alpha}\right) \le \lambda_{\mathrm{sg}}\left(d\left(v,\mathcal{F}^*\right)\right) \ , \qquad (16)$$

where the first inequality was derived in (11), and the second inequality holds by the monotonicity of λ_{sg} (Observation 3.2) and (15). Therefore, we obtain the following upper bound:

$$\sum_{v\in\mathcal{V}_1} \lambda^{\text{smooth}}_{\text{sg},\alpha}\left(\bar{d}\,(v)\right)\cdot\bar{d}\,(v) \leq \frac{\alpha}{2}\cdot\sum_{v\in\mathcal{V}_1}\lambda_{\text{sg}}\left(d\,(v,\mathcal{F}^*)\right)\cdot d\,(v,\mathcal{F}^*) \leq \frac{\alpha}{2}\cdot\psi_{\text{sg}}\left(\mathcal{F}^*\right) \leq 3\alpha\cdot\psi(\mathcal{F}^*)\ , \quad (17)$$

where the first inequality is an immediate consequence of (15) and (16), while the last inequality follows from Lemma 3.3, specialized with $\epsilon = 1$ and $\psi_{\text{sg},1} = \psi_{\text{sg}}$.

Bounding the sum over $\mathcal{V}_2$. In order to analyze the remaining terms, consider a vertex $v \in \mathcal{V}_2$. By definition of $\mathcal{V}_2$, it follows that $\bar{d}(v)/(d(v,\hat{\mathcal{F}})) \leq 1 + 1/(10\log n) \leq 1.1$. Therefore, for such vertices, $\lambda^{\text{smooth}}_{\text{sg},\alpha}(\bar{d}(v))/\lambda^{\text{smooth}}_{\text{sg},\alpha}(d(v,\hat{\mathcal{F}})) \leq n^{2/(10\log n)} \leq 1.15$, where the first inequality follows from the bounded increments of $\lambda^{\text{smooth}}_{\text{sg},\alpha}$ (Lemma 4.2). Consequently, we obtain

$$\sum_{v\in\mathcal{V}_2} \lambda^{\text{smooth}}_{\text{sg},\alpha}\left(\bar{d}\,(v)\right)\cdot\bar{d}\,(v) \leq 1.27\cdot\sum_{v\in\mathcal{V}_2}\lambda^{\text{smooth}}_{\text{sg},\alpha}\left(d\left(v,\hat{\mathcal{F}}\right)\right)\cdot d\left(v,\hat{\mathcal{F}}\right) \leq 1.27\cdot\psi^{\text{smooth}}_{\text{sg},\alpha}\left(\hat{\mathcal{F}}\right)\ . \quad (18)$$

By combining inequalities (17) and (18), we derive the desired upper bound. ∎

By inequality (14) and Lemma 4.3, we conclude that:

$$\begin{aligned} 3\cdot\psi^{\text{smooth}}_{\text{sg},\alpha}\left(\hat{\mathcal{F}}\right) &\leq \psi^{\text{smooth}}_{\text{sg},\alpha}\left(\mathcal{F}^*\right) + 6\alpha\cdot\psi(\mathcal{F}^*) + 2.54\cdot\psi^{\text{smooth}}_{\text{sg},\alpha}\left(\hat{\mathcal{F}}\right) \\ &\leq 6\,(\alpha+1)\cdot\psi\left(\mathcal{F}^*\right) + 2.54\cdot\psi^{\text{smooth}}_{\text{sg},\alpha}\left(\hat{\mathcal{F}}\right)\ , \end{aligned}$$

where the last inequality holds due to (12). It follows that $\psi^{\text{smooth}}_{\text{sg},\alpha}(\hat{\mathcal{F}}) \leq 14(\alpha+1)\cdot\psi(\mathcal{F}^*)$, meaning that $\hat{\mathcal{F}}$ is an $O(\log n)$-comparable solution for the smooth surrogate function $\psi^{\text{smooth}}_{\text{sg},\alpha}$, since $\alpha = 80\log n$. As noted in Section 4.2, this property transfers to the scaled surrogate cost, i.e., $\hat{\mathcal{F}}$ is also an $O(\log n)$-comparable solution for $\psi_{\text{sg},\alpha}$. Finally, by picking $\alpha = 80\log n$ and $\beta = 14(\alpha+1)$ in Theorem 3.4, we have just obtained an $O(\log n)$-approximation for the ordered k-median problem on general metrics.

5 Optimizing the Surrogate Cost Function on Trees

By exploiting the separability properties of the surrogate cost function $\psi_{\text{sg},}$, we argue in Section 5.1 that the surrogate problem can be solved optimally on trees using dynamic programming ideas. In turn, Lemma 3.3 implies that an optimal set of facilities for ψ_{sg} is in particular a $(1+5\epsilon)$-comparable for this function. Hence, the $(2+\epsilon)$-approximation stated in Theorem 2.2 for the ordered k-median problem on trees comes as an immediate consequence of Theorem 3.4, specialized with $\alpha = 1$ and $\beta = 1+5\epsilon$. To show that our analysis is essentially tight, we construct in Section 5.2 a sequence of tree-based instances where the optimality gap tends to $2 - O(\epsilon)$.

5.1 Dynamic program

In what follows, we present a dynamic programming formulation to optimize the surrogate function ψ_{sg} on trees in polynomial time. It is worth mentioning that an improved recursion

can be designed, with a collapsed state space of size $O(kn^2)$. However, due to the rather involved nature of the latter approach, we focus here on a simpler formulation, for ease of presentation.

Without loss of generality, we assume that the metric space (V, d) is represented by a binary edge-weighted tree $\mathcal{T} = (V, E)$. Further, we root the tree $\mathcal{T}$ at an arbitrary vertex v_0, and for any vertex $v \in V$, use $\mathcal{T}_v$ to denote the subtree rooted at v, with l_v and r_v as the left and right children of v, respectively.

State space. Each state of the dynamic program is described by the following parameters:

- $v \in V$: The root of the current subtree.
- $\kappa \in [k]_0$: The residual number of facilities to be located in $\mathcal{T}_v$.
- $u \in V \setminus (\mathcal{T}_{l_v} \cup \mathcal{T}_{r_v})$: The nearest facility to v outside of $\mathcal{T}_{l_v} \cup \mathcal{T}_{r_v}$.
- $l \in \mathcal{T}_{l_v}$: The nearest facility to v in the left subtree.
- $r \in \mathcal{T}_{r_v}$: The nearest facility to v in the right subtree.

In this setting, the function $F(v, \kappa, u, l, r)$ stands for the minimum surrogate cost due to the vertices of $\mathcal{T}_v$, over all sets of facilities $\mathcal{F} \cup \{u, l, r\}$ such that $\mathcal{F}$ is picked among $\mathcal{T}_{l_v} \cup \mathcal{T}_{r_v}$ with $l, r \in \mathcal{F}$ and $|\mathcal{F}| \leq \kappa$. When one or more of the facilities u, l, and r are not located, we use $\perp$ to denote this decision. Clearly, the optimal value of ψ_{sg} corresponds to

$$\min \left\{ \min_{l \in \mathcal{T}_{l_v}, r \in \mathcal{T}_{r_v}} F\left(v_0, k, \perp, l, r\right), \min_{l \in \mathcal{T}_{l_v}, r \in \mathcal{T}_{r_v}} F\left(v_0, k-1, v_0, l, r\right) \right\} ,$$

where the disjunction above expresses whether the root v_0 holds a facility or not.

Recursion. The recursion proceeds by first assigning the root v of the current subtree $\mathcal{T}_v$ to its nearest facility among $\{l, r, u\}$. This way, we can compute the marginal surrogate cost due to the vertex v, i.e., the quantity $\lambda_{\mathrm{sg}}(d(v, \mathcal{F})) \cdot d(v, \mathcal{F})$ with respect to the final set of facilities $\mathcal{F}$, which is precisely $\min_{x \in \{l,r,u\}} \lambda_{\mathrm{sg}}(d(v, x)) \cdot d(v, x)$.

Next, we are left with separately solving the subproblems formed by the left and right subtrees, $\mathcal{T}_{l_v}$ and $\mathcal{T}_{r_v}$. For simplicity, we discuss the case where l belongs to left subtree of $\mathcal{T}_{l_v}$ and r belongs to right subtree of $\mathcal{T}_{r_v}$; the other cases are treated through similar arguments. To update the parameter u, we define u_r as the nearest facility among $\{l, u\}$ to r_v and similarly define u_l as the nearest facility among $\{u, r\}$ to l_v. Denoting by r_0 the root of the left subtree of $\mathcal{T}_{r_v}$ and l_0 the root of the right subtree of $\mathcal{T}_{l_v}$, we obtain the following recursion:

$$\begin{aligned} F(v, k, u, l, r) \;=\; & \min_{x \in \{l,r,u\}} \lambda_{\mathrm{sg}}(d(v, x)) \cdot d(v, x) \\ & + \max_{\substack{\kappa_l \leq \kappa \\ l_1 \in \mathcal{T}_{l_0}, r_1 \in \mathcal{T}_{r_0}}} \left\{ F\left(l_v, \kappa_l, u_l, l, l_1\right) + F\left(r_v, \kappa - \kappa_l, u_r, r_1, r\right) \right\} . \end{aligned}$$

5.2 Tight example

In what follows, we show that the analysis conducted for proving Theorem 3.4 is essentially tight when the underlying metric is induced by tree, and the objective is to minimize the

surrogate cost ψ_{sg}. Specifically, given an accuracy parameter $\epsilon \in (0, 1/4)$, we construct a family of instances, indexed by an integer parameter $n \geq 25$, that matches our approximation bound up to lower-order terms, i.e., $\lim_{n\to\infty} \psi(\mathcal{F}_{\text{sg}}(n))/\psi(\mathcal{F}^*(n)) = 2-O(\epsilon)$. Here, $\mathcal{F}^*(n)$ is an optimal set of facilities for the ordered k-median problem, and $\mathcal{F}_{\text{sg}}(n)$ is an optimal surrogate solution.

We begin by constructing in Section 5.2.1 a family of worst-possible trees for the original cost function ψ. Next, we show in Section 5.2.2 that $\psi(\mathcal{F}^*(n)) = (1 + o(1)) \cdot n^2$, and prove in Section 5.2.3 that $\psi(\mathcal{F}_{\text{sg}}(n)) = (2 - 2\epsilon + o(1)) \cdot n^2$.

5.2.1 Instance construction

Graph description. The tree $\mathcal{T}(n)$ consists of three components, shown in Figure 3:

- *Core vertex.* We first introduce the core vertex C, which is connected to the auxiliary vertex A by an edge with distance $d(C, A) = 1 - \epsilon$.
- *Stars.* The core vertex C is connected to n distinct stars, indexed by $i \in [n]$. Each star is formed by a center c_i, connected to n immediate neighbors $m_1^i, \ldots, m_n^i$. Here, $d(c_i, C) = \epsilon$, and $d(c_i, m_j^i) = 1 - \epsilon$ for every neighbor index $j \in [n]$.
- *Remote vertices.* Finally, each center c_i is connected to a (distinct) remote vertex R_i by an edge with $d(c_i, R_i) = (1 - \epsilon) \cdot n$.

It is easy to verify that the tree $\mathcal{T}(n)$ consists of $n^2 + 2n + 2$ vertices.

Instance parameters. To finalize the construction, we fix the allowed number of facilities to $k = n$. In addition, the penalty weights are picked such that the top $n^2 + 1$ values are equal to 1, i.e., $\lambda_1 = \cdots = \lambda_{n^2+1} = 1$, and the $2n + 1$ remaining weights are chosen as $\lambda_{n^2+2} = \cdots = \lambda_{n^2+2n+2} = \epsilon/n^2$.

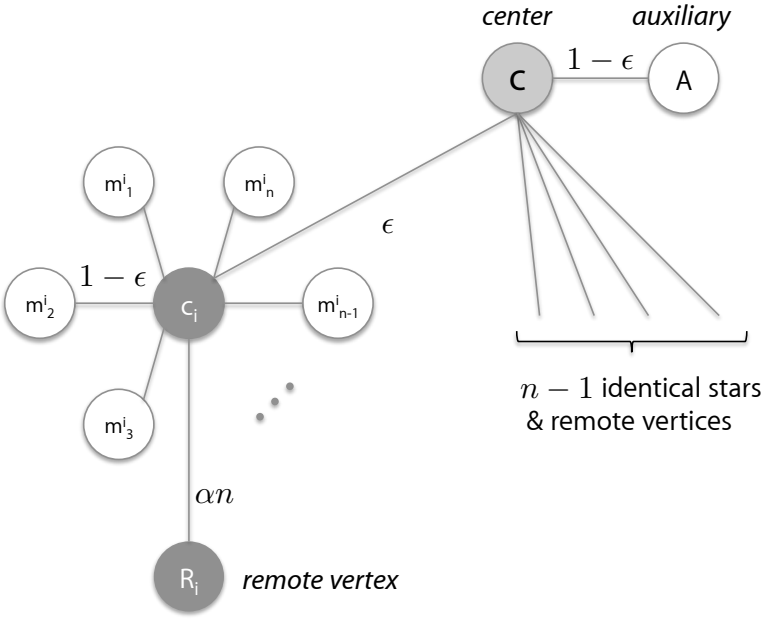

Figure 3: Illustration of the tree construction.

5.2.2 Upper bounding $\psi(\mathcal{F}^*(n))$

We begin by characterizing the set of optimal solutions to the instance constructed above.

Lemma 5.1. *Any optimal set of facilities for the ordered k-median problem is comprised of the core vertex C and $n-1$ remote vertices.*

Proof. To arrive at a contradiction, suppose that $\mathcal{F}^*$ is an optimal solution that opens at most $n-2$ facilities at the remote vertices $R_1, \ldots, R_n$. Since all pairwise distances are positive, $\mathcal{F}^*$ necessarily consists of n facilities. As a result, $\mathcal{F}^*$ contains at least two facilities at non-remote vertices. We now create a modified solution $\tilde{\mathcal{F}}$, where one of the non-remote facilities is relocated at the core vertex C, and another non-remote facility is relocated to a free remote vertex, that was not holding a facility in $\mathcal{F}^*$. To analyze the effects of this transformation, we bound the variation of the marginal cost due to each vertex $v \in V$, given by $\lambda_{i_2} \cdot d(v, \tilde{\mathcal{F}}) - \lambda_{i_1} \cdot d(v, \mathcal{F}^*)$ when v occupies the rankings i_1 in $\mathcal{F}^*$ and i_2 in $\tilde{\mathcal{F}}$. Specifically, the cost terms are broken down according to the three components of the tree $\mathcal{T}(n)$:

- *Core vertex and auxiliary vertex:* Since the core vertex C is now holding a facility in $\tilde{\mathcal{F}}$, and the auxiliary vertex A is at distance $1-\epsilon$, the variation of the cost due to these vertices is clearly upper-bounded by $\lambda_1 \cdot (1-\epsilon) = 1-\epsilon$.

- *Remote vertices:* Note that the free remote vertices in $\mathcal{F}^*$ have distances at least $(1-\epsilon)\cdot n > 2$ to their nearest facility, and therefore necessarily occupy rankings within $1, \ldots, n$, since all other non-remote vertices are within distance 2 of any non-remote facility in $\mathcal{F}^*$. As a result, their corresponding penalty weights are 1. Hence, an upper bound on the cost variation due to all remote vertices is given by $2\epsilon - (1-\epsilon) \cdot n$. Indeed, our relocation procedure increases the distance of at most two remote vertices, each by at most ϵ, and reduces to 0 the distance of at least one remote vertex (holding a new facility), incurring a cost variation of $-(1-\epsilon) \cdot n$.

- *Stars:* Our transformation relocates at most two facilities, and in addition, $\tilde{\mathcal{F}}$ holds a facility at the core vertex C, which is nearest to any star than any vertex contained in another star. Consequently, there are at most two distinct stars where the distance between a vertex to its nearest facility may increase. Within each such star, since the core vertex C holds a facility in $\tilde{\mathcal{F}}$, the distance of the two (non-remote) vertices made vacant by our transformation would increase by at most 1, while the distance of all other vertices increases by at most ϵ. In addition, there are at most 3 vertices in the stars that could have a larger penalty weight in $\tilde{\mathcal{F}}$ than in $\mathcal{F}^*$, since the only vertices outside of the star graphs with a potentially improved ranking are: the core vertex C, the auxiliary vertex A, and the remote vertex chosen for the relocation. As a result, the variation of the cost due to the stars is upper bounded by $2\epsilon n + 5$.

Overall, we obtain that $\psi(\tilde{\mathcal{F}}) - \psi(\mathcal{F}^*) \leq -(1-3\epsilon)n + 6 + \epsilon$, which is clearly negative for $n \geq 25$ and $\epsilon \in (0, 1/4)$, contradicting the optimality of $\mathcal{F}^*$.

Consequently, $\mathcal{F}^*$ opens at least $n-1$ facilities at the remote vertices $R_1, \ldots, R_n$, and it remains to show that one facility is necessarily located at the core vertex C. Specifically, the

cost of such solutions, whose sequence of ordered distances is described by (19), is

$$\varphi = (1-\epsilon)\cdot n + \epsilon + n^2 + \frac{\epsilon}{n^2}\cdot(1-\epsilon+n\epsilon) \leq n^2 + (1-\epsilon)\cdot n + 2 + \epsilon\ .$$

To complete the proof, we compare this quantity to the cost of other candidate solutions:

1. *Opening facilities at all the remote vertices* $R_1, \ldots, R_n$. This solution has a cost of at least $(1-\epsilon)\cdot n^3 \geq 3n^3/4$, by observing that the distance of the n^2 star neighbor vertices to their nearest facility is $(1-\epsilon)\cdot n + 1 - \epsilon$. Here, $\varphi < 3n^3/4$ since $n \geq 25$.

2. *Opening a facility at the auxiliary vertex* A. This solution has a cost of at least $(1-\epsilon)\cdot n + 1 + (2-\epsilon)\cdot n^2$, which is larger than the cost quantity φ since $n \geq 25$ and $\epsilon \in (0, 1/4)$.

3. *Opening a facility at a center vertex* c_i. This solution has a cost of at least $(1-\epsilon)\cdot n + (1+\epsilon)\cdot(n-1)\cdot n + 1 + (1-\epsilon)\cdot n$, which is larger than the cost quantity φ since $n \geq 25$.

4. *Opening a facility at a neighbor vertex* m^i_j. It is easy to verify that the cost of this solution is larger than that of item 3.

■

It follows that the cost of the optimal solution is $\psi(\mathcal{F}^*(n)) = \varphi = (1+o(1))\cdot n^2$

5.2.3 Lower bounding $\psi(\mathcal{F}_{\mathrm{sg}}(n))$

We now describe the surrogate cost function ψ_{sg}, arising from the construction described in Sections 3.1 and 3.2. Given the structure of optimal solutions as stated in Lemma 5.1, any such solution forms the following sequence of ordered distances:

$$(1-\epsilon)\cdot n + \epsilon, \underbrace{1, \ldots, 1}_{n^2}, 1-\epsilon, \underbrace{\epsilon, \ldots, \epsilon}_{n}, \underbrace{0, \ldots, 0}_{n}\ . \tag{19}$$

As a result, we have $\lambda_{\mathrm{sg}}((1-\epsilon)\cdot n + \epsilon) = 1$, $\lambda_{\mathrm{sg}}(1) = 1$, $\lambda_{\mathrm{sg}}(1-\epsilon) = \epsilon/n^2$, $\lambda_{\mathrm{sg}}(\epsilon) = \epsilon/n^2$, and $\lambda_{\mathrm{sg}}(d) = \epsilon/n^2$ for any $d \leq \epsilon/n$. Indeed, the preprocessing step of Section 3.1 does not modify the original penalty weights (as $i_{\min} = n^2 + n + 3$), since both the ratios between extremal penalty weights (n^2/ϵ) and between extremal positive distances ($((1-\epsilon)\cdot n + \epsilon)/\epsilon$) are smaller than $|V(\mathcal{T}(n))|/\epsilon = (n^2 + 2n + 2)/\epsilon$. We can now proceed by characterizing the optimal surrogate solution.

Lemma 5.2. *The surrogate problem has a unique optimal solution, consisting of the* n *star centers, i.e.,* $\mathcal{F}_{\mathrm{sg}}(n) = \{c_1, \ldots, c_n\}$.

Proof. To arrive at a contradiction, suppose that $\mathcal{F}$ is an optimal set of facilities to the surrogate problem, that opens at most $n-1$ facilities at the centers $c_1, \ldots, c_n$. We construct a new solution $\tilde{\mathcal{F}}$ by picking one facility $f \in \mathcal{F}$, chosen among those in $\mathcal{F} \setminus \{c_i : i \in [n]\}$ as explained below, and relocate it to a free star center, i.e., chosen out of $\{c_i : i \in [n]\} \setminus \mathcal{F}$. The proof proceeds by considering three cases.

Case 1: $\mathcal{F} \cap \{R_1, \ldots, R_n\} \neq \emptyset$. In this case, at least one star does not contain any facility in $\mathcal{F}$. Since the construction of $\mathcal{T}(n)$ is symmetric, we assume without loss of generality that the corresponding star has index 1. Consequently, f is arbitrarily picked as one of the remote vertices $\mathcal{F} \cap \{R_1, \ldots, R_n\}$, and relocated at the free center c_1. To analyze the effects of this transformation, we distinguish between remote and non-remote vertices:

- *Remote vertices:* Note that the surrogate cost terms due to remote vertices increase by at most $(1-\epsilon) \cdot n + 2\epsilon$. Indeed, the distance of f to its nearest facility in $\tilde{\mathcal{F}}$ is at most $d(f, c_1) = (1-\epsilon) \cdot n + 2\epsilon$, whereas the distance of any other remote vertex can only decrease.
- *Non-remote vertices:* The distance of any non-remote vertex to its nearest facility can only decrease following this relocation procedure, since c_1 is closer than the remote vertex f to any non-remote vertex. In particular, the distance of all immediate neighbors $m_1^1, \ldots, m_n^1$ of c_1, which was previously at least 1 since this star did not contain any facility in $\mathcal{F}$, is now $1-\epsilon$. Since $\lambda_{\text{sg}}(1-\epsilon) = \epsilon/n^2$ and $\lambda_{\text{sg}}(1) = 1$, the surrogate cost terms due to non-remote vertices decrease by at least $n \cdot (1 - \epsilon(1-\epsilon)/n^2)$.

Overall, the surrogate cost variation is bounded by

$$\psi_{\text{sg}}\left(\tilde{\mathcal{F}}\right) - \psi_{\text{sg}}\left(\mathcal{F}\right) \leq (1-\epsilon) \cdot n + 2\epsilon - n \cdot \left(1 - \frac{\epsilon}{n^2} \cdot (1-\epsilon)\right) \leq -\epsilon n + 3\epsilon \ .$$

Since $n \geq 25$, the overall variation is negative, contradicting the optimality of $\mathcal{F}$.

Case 2: $\mathcal{F} \cap \{R_1, \ldots, R_n\} = \emptyset$ and there exist i_1 and j such that $m_j^{i_1} \in \mathcal{F}$. In this case, there necessarily exists a star index i_2 (potentially equal to i_1) containing at most one facility, whose center c_{i_2} is free. Here, we create $\tilde{\mathcal{F}}$ by picking $f = m_j^{i_1}$ and relocating it to c_{i_2}. Note that, since $\epsilon \in (0, 1/4)$, this relocation may only decrease the distance of all vertices to their nearest facility, except for $m_j^{i_1}$, as their distance to c_{i_2} is smaller than that to $m_j^{i_1}$. In particular, since the star i_2 originally has at most one facility, the distance of at least $n-1$ neighbors of c_{i_2} decreases from 1 to $1-\epsilon$. On the other hand, the only increase in distance can be for $m_j^{i_1}$; however, we have $d(m_j^{i_1}, c_{i_2}) \leq d(m_j^{i_1}, c_{i_1}) + d(c_{i_1}, c_{i_2}) \leq 1 + \epsilon$. Hence, the surrogate cost variation is bounded by

$$\psi_{\text{sg}}\left(\tilde{\mathcal{F}}\right) - \psi_{\text{sg}}\left(\mathcal{F}\right) \leq (n-1) \cdot (\lambda_{\text{sg}}(1-\epsilon) - \lambda_{\text{sg}}(1)) + (1+\epsilon) = -(n-1) \cdot \left(1 - \frac{\epsilon^2}{n}\right) + 1 + \epsilon \ ,$$

which is negative for $n \geq 25$ and $\epsilon \in (0, 1/4)$, contradicting the optimality of $\mathcal{F}$.

Case 3: $\mathcal{F} \subseteq \{C, A, c_1, \ldots, c_n\}$. In this case, every star either holds a single facility at its center, or does not contain any facility. Given that facilities are either placed at the center vertices or at two other locations, namely the core vertex C and the auxiliary vertex A, and the optimal solution makes use of n facilities, there are at least $n-2$ stars holding a vertex at their center. We now pick f as a vertex in $\mathcal{F}$ that does not correspond to the center of a star (i.e., either the core vertex C or the auxiliary vertex A, since $\mathcal{F}$ has at most $n-1$ facilities in the centers $\{c_1, \ldots, c_n\}$), and relocate it to a free center c_{i_1}. As a result, the surrogate cost terms

due to the star i_1 decrease by at least

$$n \cdot (\lambda_{\mathrm{sg}}(1) - (1-\epsilon) \cdot \lambda_{\mathrm{sg}}(1-\epsilon)) \geq n \cdot \left(1 - \frac{\epsilon^2}{n}\right) .$$

On the other hand, the surrogate cost terms due to the auxiliary vertex A and the core vertex C may increase by at most 2. In addition, there is at most one additional star i_2, not holding any facility, that incurs a surrogate cost variation of at most $\epsilon(n+1)$. Indeed, the distance of its center c_{i_2} could increase by at most ϵ, since $d(c_{i_2}, f) \geq \epsilon$ and $d(c_{i_2}, c_{i_1}) = 2\epsilon$, and the distance of its neighbors $m_1^{i_2}, \ldots, m_n^{i_2}$ increases by at most ϵ as well (by the case hypothesis, the neighbor vertices do not hold a facility in $\mathcal{F}$). Finally, since $\mathcal{F}$ contains at least $n-2$ star centers, there at most two remote vertices that can be affected by this transformation, leading to a surrogate cost variation of at most 2ϵ. Overall,

$$\psi_{\mathrm{sg}}\left(\tilde{\mathcal{F}}\right) - \psi_{\mathrm{sg}}(\mathcal{F}) \leq -n \cdot \left(1 - \frac{\epsilon^2}{n}\right) + 2 + \epsilon(n+1) + 2\epsilon \leq -\frac{n}{2} + 3 .$$

Since $n \geq 25$, the surrogate cost variation is negative, contradicting the optimality of $\mathcal{F}$. ■

Based on Lemma 5.2, we can now compute the distances associated with $\mathcal{F}_{\mathrm{sg}}(n)$:

- *Auxiliary and core vertices.* The core vertex C is at distance ϵ from each of the facilities $c_1, \ldots, c_n$, while the auxiliary vertex A is at distance 1.
- *Stars.* In each star, the neighbor vertices are at distance $1-\epsilon$ from their nearest facility, located at the center, while the centers are at distance 0.
- *Remote vertices.* Each remote vertex is connected to the center of a star, and thus its nearest facility is at distance $(1-\epsilon) \cdot n$.

By arranging these distances in non-increasing order and multiplying by the penalty weights, we obtain:

$$\psi(\mathcal{F}_{\mathrm{sg}}(n)) = n \cdot (1-\epsilon) \cdot n + 1 + (n^2 - n) \cdot (1-\epsilon) + n \cdot (1-\epsilon) \cdot \frac{\epsilon}{n^2} + \epsilon \cdot \frac{\epsilon}{n^2} = (2 - 2\epsilon + o(1)) \cdot n^2 .$$

6 Concluding Remarks

Quasi-PTAS for trees. On trees metrics, some of our techniques can be utilized to obtain a $(1+\epsilon)$-approximation in time $O(n^{O(1)} \cdot k^{O((1/\epsilon)\log(n/\epsilon))})$. To this end, once the segments $\{\mathcal{D}_p\}_{p \in \mathbb{Z}}$ are defined, we guess the exact number of rankings occupied by each of the distance classes $0, \ldots, P-1$ in the optimal set of facilities $\mathcal{F}^*$, thereby obtaining the length of the intervals $\{\mathcal{I}_p^*\}$ for every $p \in [P]_0$. Next, this information is exploited by refining the dynamic program formulated in Section 5.1, where each state is now augmented with a vector $(n_0, \ldots, n_P)$, that encodes the number of rankings (or multiplicity) within each distance class of $[P]_0$. Rather than minimizing the cost function ψ, the recursion now aims at finding a certificate for feasibility, by checking whether there exists a set of facilities in the current subtree such that the multiplicity within all distance classes of $[P]_0$ is given by $(n_0, \ldots, n_P)$.

Bi-criteria approximation. Although the ordered median function ψ is not supermodular, it is not difficult to verify that, for any scaling parameter $\alpha \geq 1$, the scaled surrogate function $\psi_{\mathrm{sg},\alpha}$ satisfies this property. By leveraging recent results on the minimization of supermodular functions subject to a cardinality constraint [7], one can derive bi-criteria performance guarantees for the ordered k-median problem on general metrics, in the form of a $(1+\epsilon)$-approximation using $O(k \log n)$ facilities. It would be interesting to examine whether the capacity violation can be decreased to a constant factor (depending on $1/\epsilon$), similar to known results in this spirit for the k-median problem [31, 30, 26].

Additional open questions. In an attempt to obtain improved approximation guarantees on general metrics, one challenging direction for future research is to analyze the single-swap local search procedure with the ordered median cost function. In contrast to our surrogate methods, this approach requires dealing with the ranking intricacies of different vertices. Another interesting question is to examine whether our techniques can be refined and tailor-made to improve on the best known $O(\log n)$-approximation for the k-facility p-centrum problem [48], corresponding to the special case where $\lambda_1 = \cdots = \lambda_p = 1$ and $\lambda_{p+1} = \cdots = \lambda_n = 0$.

References

[1] I. Abraham, Y. Bartal, and O. Neiman. Nearly tight low stretch spanning trees. In *Proceedings of the 49th IEEE Annual Symposium on Foundations of Computer Science*, pages 781–790, 2008.

[2] N. Alon, R. M. Karp, D. Peleg, and D. West. A graph-theoretic game and its application to the k-server problem. *SIAM Journal on Computing*, 24(1):78–100, 1995.

[3] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit. Local search heuristics for k-median and facility location problems. *SIAM Journal on Computing*, 33(3):544–562, 2004.

[4] Y. Bartal. Probabilistic approximations of metric spaces and its algorithmic applications. In *Proceedings of the 37th IEEE Annual Symposium on Foundations of Computer Science*, pages 184–193, 1996.

[5] Y. Bartal. On approximating arbitrary metrics by tree metrics. In *Proceedings of the 30th Annual ACM Symposium on the Theory of Computing*, pages 161–168, 1998.

[6] N. Boland, P. Domínguez-Marín, S. Nickel, and J. Puerto. Exact procedures for solving the discrete ordered median problem. *Computers & Operations Research*, 33(11):3270–3300, 2006.

[7] C. Boutsidis, E. Liberty, and M. Sviridenko. Greedy minimization of weakly supermodular set functions. *arXiv preprint:1502.06528*, 2015.

[8] P. S. Bradley, U. M. Fayyad, and O. L. Mangasarian. Mathematical programming for data mining: Formulations and challenges. *INFORMS Journal on Computing*, 11(3):217–238, 1999.

[9] J. Byrka, T. Pensyl, B. Rybicki, A. Srinivasan, and K. Trinh. An improved approximation for k-median, and positive correlation in budgeted optimization. In *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 737–756, 2015.

[10] M. Charikar and S. Guha. Improved combinatorial algorithms for facility location problems. *SIAM Journal on Computing*, 34(4):803–824, 2005.

[11] M. Charikar, S. Guha, É. Tardos, and D. B. Shmoys. A constant-factor approximation algorithm for the k-median problem. In *Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, pages 1–10, 1999.

[12] P. Domínguez-Marín, S. Nickel, P. Hansen, and N. Mladenović. Heuristic procedures for solving the discrete ordered median problem. *Annals of Operations Research*, 136(1):145–173, 2005.

[13] Z. Drezner and H. W. Hamacher. *Facility Location: Applications and Theory*. Springer Science & Business Media, 2004.

[14] Z. Drezner and S. Nickel. Constructing a DC decomposition for ordered median problems. *Journal of Global Optimization*, 45(2):187–201, 2009.

[15] Z. Drezner and S. Nickel. Solving the ordered one-median problem in the plane. *European Journal of Operational Research*, 195(1):46–61, 2009.

[16] M. Elkin, Y. Emek, D. A. Spielman, and S.-H. Teng. Lower-stretch spanning trees. *SIAM Journal on Computing*, 38(2):608–628, 2008.

[17] I. Espejo, A. M. Rodríguez-Chía, and C. Valero. Convex ordered median problem with ℓ_p-norms. *Computers & Operations Research*, 36(7):2250–2262, 2009.

[18] J. Fakcharoenphol, S. Rao, and K. Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *Journal of Computer and System Sciences*, 69(3):485–497, 2004.

[19] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998.

[20] E. Gassner. An inverse approach to convex ordered median problems in trees. *Journal of Combinatorial Optimization*, 23(2):261–273, 2012.

[21] A. Gupta and K. Tangwongsan. Simpler analyses of local search algorithms for facility location. *arXiv preprint:0809.2554*, 2008.

[22] D. S. Hochbaum and D. B. Shmoys. A best possible heuristic for the k-center problem. *Mathematics of Operations Research*, 10(2):180–184, 1985.

[23] K. Jain and V. V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *Journal of the ACM*, 48(2):274–296, 2001.

[24] J. Kalcsics, S. Nickel, and J. Puerto. Multifacility ordered median problems on networks: A further analysis. *Networks*, 41(1):1–12, 2003.

[25] J. Kalcsics, S. Nickel, J. Puerto, and A. Tamir. Algorithmic results for ordered median problems. *Operations Research Letters*, 30(3):149–158, 2002.

[26] M. R. Korupolu, C. G. Plaxton, and R. Rajaraman. Analysis of a local search heuristic for facility location problems. *Journal of Algorithms*, 37(1):146–188, 2000.

[27] G. Laporte, S. Nickel, and F. S. da Gama. *Location Science*. Springer, 2015.

[28] T. L. Lei and R. L. Church. Vector assignment ordered median problem: A unified median problem. *International Regional Science Review*, 37(2):194–224, 2014.

[29] S. Li and O. Svensson. Approximating k-median via pseudo-approximation. *SIAM Journal on Computing*, 45(2):530–547, 2016.

[30] J.-H. Lin and J. S. Vitter. Approximation algorithms for geometric median problems. *Information Processing Letters*, 44(5):245–249, 1992.

[31] J.-H. Lin and J. S. Vitter. e-approximations with minimum packing constraint violation. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, pages 771–782, 1992.

[32] A. J. Lozano and F. Plastria. The ordered median Euclidean straight-line location problem. *Studies in Locational Analysis*, 17:29–43, 2009.

[33] M. Mendel and A. Naor. Maximum gradient embeddings and monotone clustering. *Combinatorica*, 30(5):581–615, 2010.

[34] P. B. Mirchandani and R. L. Francis. *Discrete Location Theory*. Wiley, 1990.

[35] S. Nickel. Discrete ordered weber problems. In *Operations Research Proceedings: Selected Papers of the Symposium on Operations Research (OR 2000)*, pages 71–76, 2001.

[36] S. Nickel and J. Puerto. A unified approach to network location problems. *Networks*, 34(4):283–290, 1999.

[37] S. Nickel and J. Puerto. *Location Theory: A Unified Approach*. Springer Science & Business Media, 2005.

[38] J. Puerto and F. R. Fernández. Geometrical properties of the symmetrical single facility location problem. *Journal of Nonlinear and Convex Analysis*, 1(3):321–342, 2000.

[39] J. Puerto, D. Pérez-Brito, and C. G. García-González. A modified variable neighborhood search for the discrete ordered median problem. *European Journal of Operational Research*, 234(1):61–76, 2014.

[40] J. Puerto, A. B. Ramos, and A. M. Rodríguez-Chía. Single-allocation ordered median hub location problems. *Computers & Operations Research*, 38(2):559–570, 2011.

[41] J. Puerto, A. B. Ramos, and A. M. Rodríguez-Chía. A specialized branch & bound & cut for single-allocation ordered median hub location problems. *Discrete Applied Mathematics*, 161(16):2624–2646, 2013.

[42] J. Puerto and A. M. Rodríguez-Chía. On the exponential cardinality of FDS for the ordered p-median problem. *Operations Research Letters*, 33(6):641–651, 2005.

[43] J. Puerto, A. M. Rodríguez-Chía, and A. Tamir. Minimax regret single-facility ordered median location problems on networks. *INFORMS Journal on Computing*, 21(1):77–87, 2009.

[44] J. Puerto, A. M. Rodríguez-Chía, and A. Tamir. Revisiting k-sum optimization. *Mathematical Programming*, pages 1–26, 2016.

[45] J. Puerto and A. Tamir. Locating tree-shaped facilities using the ordered median objective. *Mathematical Programming*, 102(2):313–338, 2005.

[46] A. M. Rodríguez-Chía, J. Puerto, D. Pérez-Brito, and J. A. Moreno. The p-facility ordered median problem on networks. *Top*, 13(1):105–126, 2005.

[47] Z. Stanimirović, J. Kratica, and D. Dugošija. Genetic algorithms for solving the discrete ordered median problem. *European Journal of Operational Research*, 182(3):983–1001, 2007.

[48] A. Tamir. The k-centrum multi-facility location problem. *Discrete Applied Mathematics*, 109(3):293–307, 2001.

[49] A. Tamir, D. Pérez-Brito, and J. A. Moreno-Pérez. A polynomial algorithm for the p-centdian problem on a tree. *Networks*, 32(4):255–262, 1998.

A Inapproximability of the Surrogate Problem

In what follows, we prove that the surrogate model is strongly inapproximable. Here, we are given a finite metric space (V, d) on n vertices, a non-decreasing left-continuous step-function $\lambda : [0, \infty) \to \mathbb{R}^+$, and an integer parameter k. For any set of facilities $\mathcal{F} \subseteq V$, the distance $d(V, \mathcal{F})$ of each vertex $v \in V$ to its nearest facility is penalized by a multiplicative weight of $\lambda(d(v, \mathcal{F}))$. The objective is to compute a set $\mathcal{F} \subseteq V$ of at most k facilities that minimizes the surrogate function $\psi_{\text{sg}}(\mathcal{F}) = \sum_{v \in V} \lambda(d(v, \mathcal{F})) \cdot d(v, \mathcal{F})$.

Theorem A.1. *There are constants $\delta \in (0, 1)$ and $C_{\text{sg}} > 0$ such that the surrogate model cannot be approximated in polynomial time within factor $C_{\text{sg}} n^{\delta} \ln n$, unless* $\text{NP} \subseteq \text{TIME}(n^{O(\log \log n)})$.

Proof. To establish the claim, we describe a gap-preserving reduction from the dominating set problem. Given an undirected graph $G = (V, E)$, a subset of vertices $D \subseteq V$ is a dominating set if every vertex not in D is adjacent to at least one member of D. The objective is to compute a minimum-cardinality dominating set. We utilize a well-known inapproximability result of Feige [19], stating that the set cover problem cannot be efficiently approximated within factor

$(1-\epsilon)\cdot\ln n$ for any $\epsilon \in (0,1)$, unless NP $\subseteq$ TIME($n^{O(\log\log n)}$). When this result is translated to dominating set terms, including the precise parameters involved in Feige's construction, one can infer that there exists a constant $\delta \in (0,1)$ such that it is hard to distinguish between graphs with $\gamma(G) \leq n^{\delta}$ and those with $\gamma(G) \geq n^{\delta}\cdot(1-\epsilon)\cdot\ln n$ under the same complexity assumption, where $\gamma(G)$ stands for the minimum cardinality of a dominating set in G.

Now, given an instance of the dominating set problem, consisting of a graph $G=(V,E)$ on n vertices, we define a corresponding instance of the surrogate model as follows:

- The underlying metric (V,d) on the same set of vertices is obtained by defining the distance function:
$$d(u,v) = \begin{cases} 1, & \text{if } (u,v)\in E \\ 2, & \text{if } (u,v)\notin E\ . \end{cases}$$
- The penalty function $\lambda : [0,\infty) \to \mathbb{R}^{+}$ is given by
$$\lambda(d) = \begin{cases} 1, & \text{if } d\in[0,1.5) \\ n, & \text{if } d\in[1.5,\infty)\ . \end{cases}$$
- The number of facilities to be located is at most $k = n^{\delta}$.

We first argue that $\gamma(G) \leq n^{\delta}$ implies the existence of a feasible facility set $\mathcal{F}$ with $\psi_{\text{sg}}(\mathcal{F}) \leq n$. Indeed, by picking $\mathcal{F}$ as a minimum-cardinality dominating set, every remaining vertex is within distance 1 of its nearest facility in $\mathcal{F}$, and therefore, $\psi_{\text{sg}}(\mathcal{F}) = n - |\mathcal{F}| \leq n$. Conversely, when $\gamma(G) \geq n^{\delta}\cdot(1-\epsilon)\cdot\ln n$, any set of at most n^{δ} facilities leaves at least $n^{\delta}\cdot(1-\epsilon)\cdot\ln n - n^{\delta} - 1$ vertices whose distance to their nearest facility is 2. Otherwise, by adding these vertices to the chosen set of facilities, we would have obtained a dominating set of size smaller than $n^{\delta}\cdot(1-\epsilon)\cdot\ln n$. As a result, for the optimal set of facilities $\mathcal{F}^{*}$, we must have $\psi_{\text{sg}}(\mathcal{F}^{*}) \geq (n^{\delta}\cdot(1-\epsilon)\cdot\ln n - n^{\delta} - 1)\cdot n$.

To summarize, it follows that unless NP $\subseteq$ TIME($n^{O(\log\log n)}$), the surrogate model cannot be approximated in polynomial time within factor $n^{\delta}\cdot(1-\epsilon)\cdot\ln n - n^{\delta} - 1 \geq \frac{1-\epsilon}{2}\cdot n^{\delta}\ln n$, for sufficiently large n. ■

It is worth mentioning that our reduction creates instances of the surrogate model where the ratio between the maximum and minimum values of λ_{sg} is $O(n)$, similar to the instances created by the algorithm we present in Section 3.1.

B Additional Proofs

B.1 Proof of Lemma 3.1

Since only the penalty weights of rankings $i \geq i_{\min}$ are modified, we have

$$\begin{aligned} \tilde{\psi}(\mathcal{F}^{*}) - \psi(\mathcal{F}^{*}) &= \sum_{i=i_{\min}}^{n} (\tilde{\lambda}_i - \lambda_i)\cdot\Delta_{\mathcal{F}^{*}}(i) \\ &\leq n\cdot\max_{i_{\min}\leq i\leq n}\left\{(\tilde{\lambda}_i - \lambda_i)\cdot\Delta_{\mathcal{F}^{*}}(i)\right\} \end{aligned}$$

$$\begin{aligned} &\leq \epsilon \cdot \lambda_1 \cdot \Delta_{\mathcal{F}^*}(1) \\ &\leq \epsilon \cdot \psi(\mathcal{F}^*) \ , \end{aligned}$$

where the second inequality holds by observing that for any ranking $i \geq i_{\min}$ we have $\tilde{\lambda}_i \leq \frac{\epsilon \cdot \lambda_1}{n}$ or $\Delta_{\mathcal{F}^*}(i) \leq \frac{\epsilon \cdot \Delta_{\mathcal{F}^*}(1)}{n}$, while $\lambda_i \leq \lambda_1$ and $\Delta_{\mathcal{F}^*}(i) \leq \Delta_{\mathcal{F}^*}(1)$.

B.2 Proof of Claim 2 in Lemma 3.5

Consider some $i \in [n]$ for which $\text{sign}(i) = +$. By definition of the sign function, we know that $\Delta_{\mathcal{F}}(i) \leq \alpha \cdot \Delta_{\mathcal{F}^*}(i)/(1+\epsilon)$, meaning that:

$$\lambda_{\text{sg},\alpha}\left(\Delta_{\mathcal{F}}(i)\right) = \lambda_{\text{sg}}\left(\frac{\Delta_{\mathcal{F}}(i)}{\alpha}\right) \leq \lambda_{\text{sg}}\left(\frac{\Delta_{\mathcal{F}^*}(i)}{1+\epsilon}\right) \ ,$$

where the latter inequality follows from the monotonicity of λ_{sg} (Observation 3.2). We now distinguish between three cases.

Case 1: $\boldsymbol{\pi^*(i) = P}$. Here, we simply observe that λ_{sg} is constant across all distance classes $p \geq P$. Therefore,

$$\lambda_{\text{sg}}\left(\frac{\Delta_{\mathcal{F}^*}(i)}{1+\epsilon}\right) = \lambda_{\text{sg}}\left(\Delta_{\mathcal{F}^*}(i)\right) = \lambda_{\text{guess}}(P) \leq (1+\epsilon) \cdot \lambda_{\text{avg}}(P) = (1+\epsilon) \cdot \lambda_i \ ,$$

where the third equality holds since $i \in \mathcal{I}^*_P$. Indeed, it implies that $i \geq i_{\min}$ by definition of $i_{\min}$, while $\lambda_{i_{\min}} = \cdots = \lambda_n$ due to our preprocessing step (see Section 3.1).

Case 2: $\boldsymbol{\pi^*(i) \leq P-1}$ and $\boldsymbol{\mathcal{I}_{\pi^*(i)+1} = \emptyset}$. In this case, $\Delta_{\mathcal{F}^*}(i)/(1+\epsilon) \in \mathcal{D}_{\pi^*(i)+1}$. Since $\mathcal{I}_{\pi^*(i)+1}$ is empty, by observing that $\mathcal{I}_{\pi^*(i)}$ is the first non-empty interval located at the left of $\mathcal{I}_{\pi^*(i)+1}$ (as $i \in \mathcal{I}_{\pi^*(i)}$), it follows that

$$\lambda_{\text{sg}}\left(\frac{\Delta_{\mathcal{F}^*}(i)}{1+\epsilon}\right) = \lambda_{\text{guess}}\left(\pi^*(i)+1\right) \leq (1+\epsilon) \cdot \lambda_{\text{avg}}\left(\pi^*(i)+1\right) \leq (1+\epsilon) \cdot \lambda_i \ ,$$

where the last inequality holds since $i \in \mathcal{I}_{\pi^*(i)}$ and $\lambda_1 \geq \cdots \geq \lambda_n$.

Case 3: $\boldsymbol{\pi^*(i) \leq P-1}$ and $\boldsymbol{\mathcal{I}_{\pi^*(i)+1} \neq \emptyset}$. Here, $\Delta_{\mathcal{F}^*}(i)/(1+\epsilon) \in \mathcal{D}_{\pi^*(i)+1}$ as well, and we obtain:

$$\begin{aligned} \lambda_{\text{sg}}\left(\frac{\Delta_{\mathcal{F}^*}(i)}{1+\epsilon}\right) &= \lambda_{\text{guess}}\left(\pi^*(i)+1\right) \\ &\leq (1+\epsilon) \cdot \lambda_{\text{avg}}\left(\pi^*(i)+1\right) \\ &= \frac{1+\epsilon}{|\mathcal{I}^*_{\pi^*(i)+1}|} \cdot \sum_{j \in \mathcal{I}^*_{\pi^*(i)+1}} \lambda_j \\ &\leq (1+\epsilon) \cdot \lambda_i \ . \end{aligned}$$

The first inequality follows from the accuracy of the guessing procedure (see inequality (2)), while the last inequality holds given that $\lambda_1 \geq \cdots \geq \lambda_n$ and that the ranking i is located at the

left of the interval $\mathcal{I}^*_{\pi^*(i)+1}$ (since $i \in \mathcal{I}^*_{\pi^*(i)}$).

B.3 Proof of Lemma 4.2

Let $p \geq p'$ be the unique integers for which $d \in \mathcal{D}^\alpha_p$ and $d' \in \mathcal{D}^\alpha_{p'}$. We consider three cases, depending on the relation between p and p'.

- $\boldsymbol{p' \leq p-2}$. In this case,

$$\frac{\lambda^{\text{smooth}}_{\text{sg},\alpha}(d')}{\lambda^{\text{smooth}}_{\text{sg},\alpha}(d)} \leq \frac{\lambda_1}{\lambda_n} \leq n \leq n^{2(d'/d-1)} \ ,$$

  where the last inequality holds since $d'/d \geq (\min \mathcal{D}^\alpha_{p'})/(\max \mathcal{D}^\alpha_p) = 2^{p-p'-1} \geq 2$.

- $\boldsymbol{p' = p-1}$. In this case, letting $d = \gamma \cdot \delta_p/2$ and $d' = \gamma' \cdot \delta_{p-1}/2$, we obtain

$$\frac{\lambda^{\text{smooth}}_{\text{sg},\alpha}(d')}{\lambda^{\text{smooth}}_{\text{sg},\alpha}(d)} = \eta_p^{2-\gamma} \cdot \eta_{p-1}^{\gamma'-1} \leq n^{(\gamma'-1+\gamma\cdot(2/\gamma-1))} \leq n^{2(\gamma'-1+2/\gamma-1)} \leq n^{2(2\gamma'/\gamma-1)} = n^{2(d'/d-1)} \ .$$

  Here, the first inequality is due to $\max\{\eta_p, \eta_{p-1}\} \leq n$, and the second inequality holds since $\gamma \leq 2$. The third inequality proceeds from observing that $\gamma' \geq 1$ and $2/\gamma \geq 1$, thus $\gamma' \cdot (2/\gamma) - 1 \geq (\gamma' - 1) + (2/\gamma - 1)$. Finally, the last equality holds since $d'/d = (\gamma' \cdot \delta_{p-1})/(\gamma \cdot \delta_p) = 2\gamma'/\gamma$.

- $\boldsymbol{p' = p}$. Letting $d = \gamma \cdot \delta_p/2$ and $d' = \gamma' \cdot \delta_p/2$, we have

$$\frac{\lambda^{\text{smooth}}_{\text{sg},\alpha}(d')}{\lambda^{\text{smooth}}_{\text{sg},\alpha}(d)} = \eta_p^{\gamma'-\gamma} = \eta_p^{\gamma\cdot(d'/d-1)} \leq n^{2(d'/d-1)} \ ,$$

  where the last inequality holds since $\eta_p \leq n$ and $\gamma \leq 2$.