
Facility Location Problem in Differential Privacy Model Revisited

Yunus Esencayi*
SUNY at Buffalo

Marco Gaboardi
Boston University

Shi Li
SUNY at Buffalo

Di Wang
SUNY at Buffalo

Abstract

In this paper we study the uncapacitated facility location problem in the model of differential privacy (DP) with uniform facility cost. Specifically, we first show that, under the *hierarchically well-separated tree (HST) metrics* and the *super-set output setting* that was introduced in [8], there is an ϵ -DP algorithm that achieves an $O(\frac{1}{\epsilon})$ (expected multiplicative) approximation ratio; this implies an $O(\frac{\log n}{\epsilon})$ approximation ratio for the general metric case, where n is the size of the input metric. These bounds improve the best-known results given by [8]. In particular, our approximation ratio for HST-metrics is independent of n , and the ratio for general metrics is independent of the aspect ratio of the input metric.

On the negative side, we show that the approximation ratio of any ϵ -DP algorithm is lower bounded by $\Omega(\frac{1}{\sqrt{\epsilon}})$, even for instances on HST metrics with uniform facility cost, under the super-set output setting. The lower bound shows that the dependence of the approximation ratio for HST metrics on ϵ can not be removed or greatly improved. Our novel methods and techniques for both the upper and lower bound may find additional applications.

1 Introduction

The facility location problem is one of the most fundamental problems in combinatorial optimization and has a wide range of applications such as plant or warehouse location problems and network design problems, also it is closely related to clustering problems such as k -median, where one typically seeks to partition a set of data points, which themselves find applications in data mining, machine learning, and bioinformatics [1, 13, 4]. Due to its versatility, the problem has been studied by both operations research and computer science communities [20, 19, 16, 15, 1, 13, 4]. Formally, it can be defined as following.

Definition 1 (Uniform Facility Location Problem (Uniform-FL)). The input to the Uniform Facility Location (Uniform-FL) problem is a tuple (V, d, f, \vec{N}) , where (V, d) is a n -point discrete metric, $f \in \mathbb{R}_{\geq 0}$ is the facility cost, and $\vec{N} = (N_v)_{v \in V} \in \mathbb{Z}_{\geq 0}^V$ gives the number of clients in each location $v \in V$. The goal of the problem is to find a set of facility locations $S \subseteq V$ which minimize the following, where $d(v, S) = \min_{s \in S} d(v, s)$,

$$\min_{S \subseteq V} \text{cost}_d(S; \vec{N}) := |S| \cdot f + \sum_{v \in V} N_v d(v, S). \quad (1)$$

The first term of (1) is called the *facility cost* and the second term is called the *connection cost*.

Throughout the paper, we shall simply use **UFL** to refer to Uniform-FL. Although the problem has been studied quite well in recent years, there is some privacy issue on the locations of the clients. Consider the following scenario: One client may get worried that the other clients may be able to obtain some information on her location by colluding and exchanging their information. As a

*Authors are alphabetically ordered.

commonly-accepted approach for preserving privacy, Differential Privacy (DP) [5] provides provable protection against identification and is resilient to arbitrary auxiliary information that might be available to attackers.

However, under the ϵ -DP model, Gupta et al. [8] recently showed that it is impossible to achieve a useful multiplicative approximation ratio of the facility location problem. Specifically, they showed that any 1-DP algorithm for UFL under general metric that outputs the set of open facilities must have a (multiplicative) approximation ratio of $\Omega(\sqrt{n})$ which negatively show that UFL in DP model is useless. Thus one needs to consider some relaxed settings in order to address the issue.

In the same paper [8] the authors showed that, under the following setting, an $O(\frac{\log^2 n \log^2 \Delta}{\epsilon})$ approximation ratio under the ϵ -DP model is possible, where $\Delta = \max_{u,v \in V} d(u,v)$ is the diameter of the input metric. In the setting, the output is a set $R \subseteq V$, which is a *super-set* of the set of open facilities. Then every client sees the output R and chooses to connect to its nearest facility in R . The actual set S of open facilities, is the facilities in R with at least one connecting client. **Thus, in this model, a client will only know its own service facility, instead of the set of open facilities.** We call this setting the *super-set output setting*. Roughly speaking, under the ϵ -DP model, one can not well distinguish between if there is 0 or 1 client at some location v . If v is far away from all the other locations, then having 1 at v will force the algorithm to open v and thus will reveal information about the existence of the client at v . This is how the lower bound in [8] was established. By using the super-set output setting, the algorithm can always output v and thus does not reveal much information about the client. If there is no client at v then v will not be open.

In this paper we further study the UFL problem in the ϵ -DP model with the super-set output setting by [8] we address the following questions.

For the UFL problem under the ϵ -DP model and the super-set output setting, can we do better than the results in [8] in terms of the approximation ratio? Also, what is the lower bound of the approximation ratio in the same setting?

We make progresses on both problems. Our contributions can be summarized as the followings.

- We show that under the so called *Hierarchical-Well-Separated-Tree (HST)* metrics, there is an algorithm that achieves $O(\frac{1}{\epsilon})$ approximation ratio. By using the classic FRT tree embedding technique of [6], we can achieve $O(\frac{\log n}{\epsilon})$ approximation ratio for any metrics, under the ϵ -DP model and the super-set output setting. These factors respectively improve upon a factor of $O(\log n \log^2 \Delta)$ in [8] for HST and general metrics. Thus, for HST-metrics, our approximation only depends on ϵ . For general metrics, our result removed the poly-logarithmic dependence on Δ in [8].
- On the negative side, we show that the approximation ratio under ϵ -DP model is lower bounded by $\Omega(\frac{1}{\sqrt{\epsilon}})$ even if the metric is a star (which is a special case of a HST). This shows that the dependence on ϵ is unavoidable and can not be improved greatly.

Due to space limit, some proofs are deferred to the supplementary material.

Related Work The work which is the most related to this paper is [8], where the author first studied the problem. Nissim et al. [18] study an abstract mechanism design model where DP is used to design approximately optimal mechanism, and they use facility location as one of their key examples. The UFL problem has close connection to k -median clustering and submodular optimization, whose DP versions have been studied before such as [17, 3, 7, 8, 2]. However, their methods cannot be used in our problem. There are many papers study other combinatorial optimization problems in DP model such as [9, 10, 11, 12, 8].

Finally, we remark that the setting we considered in the paper is closely related to the *Joint Differential Privacy Model* that was introduced in [14]. We leave the details to the supplementary material.

2 Preliminaries

Giving a data universe V and a dataset $D = \{v_1, \dots, v_N\} \in V^N$ where each record v_i belongs to an individual i whom we refer as a client in this paper. Let $\mathcal{A} : V^N \mapsto \mathcal{S}$ be an algorithm on D and produce an output in \mathcal{S} . Let D_{-i} denote the dataset D without entry of the i -th client. Also denote (v'_i, D_{-i}) denote the dataset by adding v'_i to D_{-i} .

Definition 2 (Differential Privacy [5]). A randomized algorithm \mathcal{A} is ϵ -differentially private (DP) if for any client $i \in [N]$, any two possible data entries $v_i, v'_i \in V$, any dataset $D_{-i} \in V^{N-1}$ and for all events $\mathcal{T} \subseteq \mathcal{S}$ in the output space of \mathcal{A} , we have $\Pr[\mathcal{A}(v_i, D_{-i}) \in \mathcal{T}] \leq e^\epsilon \Pr[\mathcal{A}(v'_i, D_{-i}) \in \mathcal{T}]$.

For the UFL problem, instead of using a set D of clients as input, it is more convenient for us to use a vector $\vec{N} = (N_v)_{v \in V} \in \mathbb{Z}_{\geq 0}^V$, where N_v indicates the number of clients at location v . Then the ϵ -DP requires that for any input vectors \vec{N} and \vec{N}' with $|\vec{N} - \vec{N}'|_1 = 1$ and any event $\mathcal{T} \subseteq \mathcal{S}$, we have $\Pr[\mathcal{A}(\vec{N}) \in \mathcal{T}] \leq e^\epsilon \Pr[\mathcal{A}(\vec{N}') \in \mathcal{T}]$.

In the super-set output setting for the UFL problem, the output of an algorithm is a set $R \subseteq V$ of potential open facilities. Then, every client, or equivalently, every location v with $N_v \geq 1$, will be connected to the nearest location in R under some given metric (in our algorithm, we use the HST tree metric). Then the actual set S of open facilities is the set of locations in R with at least 1 connected client. Notice that the connection cost of S will be the same as that of R ; but the facility cost might be much smaller than that of R . This is why the super-set output setting may help in getting good approximation ratios.

Throughout the paper, approximation ratio of an algorithm \mathcal{A} is the *expected multiplicative approximation ratio*, which is the expected cost of the solution given by the algorithm, divided by the cost of the optimum solution, i.e., $\frac{\mathbb{E}\text{cost}_d(\mathcal{A}(\vec{N}); \vec{N})}{\min_{S \subseteq V} \text{cost}_d(S; \vec{N})}$, where the expectation is over the randomness of \mathcal{A} .

Organization In Section 3, we show how to reduce UFL on general metrics to that on HST metrics, while losing a factor of $O(\log n)$ in the approximation ratio. In Section 4, we give our ϵ -DP $O(1/\epsilon)$ -approximation for UFL under the super-set output setting. Finally in Section 5, we prove our $\Omega(1/\sqrt{\epsilon})$ -lower bound on the approximation ratio for the same setting.

3 Reducing General Metrics to Hierarchically Well-Separated Tree Metrics

The classic result of Fakcharoenphol, Rao and Talwar (FRT) [6] shows that any metric on n points can be embedded into a distribution of metrics induced by *hierarchically well-separated trees* with distortion $O(\log n)$. As in [8], this tree-embedding result is our starting point for our DP and LDP algorithms for uniform UFL. To apply the technique, we first define what is a hierarchically well-separated tree.

Definition 3. For any real number $\lambda > 1$, an integer $L \geq 1$, a λ -Hierarchically Well-Separated tree (λ -HST) of depth L is an edge-weighted rooted tree T satisfying the following properties:

- (3a) Every root-to-leaf path in T has exactly L edges.
- (3b) If we define the level of a vertex v in T to be L minus the number of edges in the unique root-to- v path in T , then an edge between two vertices of level ℓ and $\ell + 1$ has weight λ^ℓ .

Given a λ -HST T , we shall always use V_T to denote its vertex set. For a vertex $v \in V_T$, we let $\ell_T(v)$ denote the *level of v* using the definition in (3b). Thus, the root r of T has level $\ell_T(r) = L$ and every leaf $v \in T$ has level $\ell_T(v) = 0$. For every $u, v \in V_T$, define $d_T(u, v)$ be the total weight of edges in the unique path from u to v in T . So (V_T, d_T) is a metric. With the definitions, we have:

Fact 1. Let $u \in V_T$ be a non-leaf of T and $v \neq u$ be a descendant of u , then

$$\lambda^{\ell_T(u)-1} \leq d_T(u, v) \leq \frac{\lambda^{\ell_T(u)-1}}{\lambda-1} \leq \frac{\lambda^{\ell_T(u)}}{\lambda-1}.$$

We say a metric (V, d) is a λ -HST metric for some $\lambda > 1$ if there exists a λ -HST T with leaves being V such that $(V, d) \equiv (V, d_T|_V)$, where $d_T|_V$ is the function d_T restricted to pairs in V . Throughout the paper, we guarantee that if a metric is a λ -HST metric, the correspondent λ -HST T is given. We give the formal description of the FRT result as well how to apply it to reduce UFL on general metrics to that on $O(1)$ -HST metrics in the supplementary material. Specifically, we shall prove the following theorem:

Theorem 1. Let $\lambda > 1$ be any absolute constant. If there exists an efficient ϵ -DP $\alpha_{\text{tree}}(n, \epsilon)$ -approximation algorithm \mathcal{A} for UFL on λ -HST's under the super-set output setting, then there exists an efficient ϵ -DP $O(\log n) \cdot \alpha_{\text{tree}}(n, \epsilon)$ -approximation algorithm for UFL on general metrics under the same setting.

In Section 4, we shall show that it is possible to make $\alpha_{\text{tree}}(n, \epsilon) = O(1/\epsilon)$:

Theorem 2. For every small enough absolute constant $\lambda > 1$, there is an efficient ϵ -DP $O(1/\epsilon)$ -approximation algorithm for UFL on λ -HST metrics under the super-set output setting.

Combining Theorems 1 and 2 will give our main theorem.

Theorem 3 (Main Theorem). Given any UFL tuple (V, d, f, \vec{N}) where $|V| = n$ and $\epsilon > 0$, there is an efficient ϵ -DP algorithm \mathcal{A} in the super-set output setting achieving an approximation ratio of $O(\frac{\log n}{\epsilon})$.

4 ϵ -DP Algorithm with $O(1/\epsilon)$ Approximation Ratio for HST Metrics

In this section, we prove Theorem 2. Let $\lambda \in (1, 2)$ be any absolute constant and let $\eta = \sqrt{\lambda}$. We prove the theorem for this fixed λ . So we are given a λ -HST T with leaves being V . Our goal is to design an ϵ -DP ($\alpha_{\text{tree}} = O(1/\epsilon)$)-approximation algorithm for UFL instances on the metric (V, d_T) . Our input vector is $\vec{N} = (N_v)_{v \in V}$, where $N_v \in \mathbb{Z}_{\geq 0}$ is the number of clients at the location $v \in V$.

4.1 Useful Definitions and Tools

Before describing our algorithm, we introduce some useful definitions and tools. Recall that V_T is the set of vertices in T and $V \subseteq V_T$ is the set of leaves. Since we are dealing with a fixed T in this section, we shall use $\ell(v)$ for $\ell_T(v)$. Given any $u \in V_T$, we use T_u to denote the sub-tree of T rooted at u . Let $L \geq 1$ be the depth of T ; we assume $L \geq \log_{\lambda}(\epsilon f)$.² We use $L' = \max\{0, \lceil \log_{\lambda}(\epsilon f) \rceil\} \leq L$ to denote the smallest non-negative integer ℓ such that $\lambda^{\ell} \geq \epsilon f$.

We extend the definition of N_u 's to non-leaves u of T : For every $u \in V_T \setminus V$, let $N_u = \sum_{v \in T_u \cap V} N_v$ to be the total number of clients in the tree T_u . We can assume that facilities can be built at any location $v \in V_T$ (instead of only at leaves V): On one hand, this assumption enriches the set of valid solutions and thus only decreases the optimum cost. On the other hand, for any $u \in V_T$ with an open facility, we can move the facility to any leaf v in T_u . Then for any leaf $v' \in V$, it is the case that $d(v', v) \leq 2d(v', u)$. Thus moving facilities from $V_T \setminus V$ to V only incurs a factor of 2 in the connection cost.

An important function that will be used throughout this section is the following set of *minimal vertices*:

Definition 4. For a set $M \subseteq T$ of vertices in T , let

$$\text{min-set}(M) := \{u \in M : \forall v \in T_u \setminus \{u\}, v \notin M\}.$$

For every v , let we define $B_v := \min\{f, N_v \lambda^{\ell(v)}\}$. This can be viewed as a lower bound on the cost incurred inside the tree T_v , as can be seen from the following claim:

Claim 1. Let $V' \subseteq V_T$ be a subset of vertices that does not contain an ancestor-descendant pair³. Then we have $\text{opt} \geq \sum_{v \in V'} B_v$.

4.2 Base Algorithm for UFL without Privacy Guarantee

Before describing the ϵ -DP algorithm, we first give a base algorithm (Algorithm 1) without any privacy guarantee as the starting point of our algorithmic design. The algorithm gives an approximation ratio of $O(1/\epsilon)$; however, it is fairly simple to see that by making a small parameter change, we can achieve $O(1)$ -approximation ratio. We choose to present the algorithm with $O(1/\epsilon)$ -ratio only to make it closer to our final algorithm (Algorithm 2), which is simply the noise-version of the base algorithm. The noise makes the algorithm ϵ -DP, while only incurring a small loss in the approximation ratio.

Recall that we are considering the super-set output setting, where we return set R of facilities, but only open a set $S \subseteq R$ of facilities using the following *closest-facility rule*: We connect every client to its nearest facility in R , then the set $S \subseteq R$ of open facilities is the set of facilities in R with at least 1 connected client.

²If this is not the case, we can repeat the following process many steps until the condition holds: create a new root for T and let the old root be its child.

³This means for every two distinct vertices $u, v \in V'$, u is not an ancestor of v

Algorithm 1 UFL-tree-base(ϵ)

- 1: Let $M \leftarrow \left\{ v \in V_T : \ell(v) \geq L' \text{ or } N_v \cdot \lambda^{\ell(v)} \geq f \right\}$ be the set of marked vertices
 - 2: $R \leftarrow \text{min-set}(M)$
 - 3: **return** R but only open $S \subseteq R$ using the closest-facility rule.
-

In the base algorithm, M is the set of *marked* vertices in T and we call vertices not in T *unmarked*. All vertices at levels $[L', L]$ are marked. Notice that there is a monotonicity property among vertices in V_T : for two vertices $u, v \in V_T$ with u being an ancestor of v , v is marked implies that u is marked. Due to this property, we call an unmarked vertex $v \in V_T$ *maximal-unmarked* if its parent is marked. Similarly, we call a marked vertex $v \in V_T$ *minimal-marked* if all its children are unmarked (this is the case if v is a leaf). So R is the set of minimal-marked vertices. Notice one difference between our algorithm and that of [8]: we only return minimal-marked vertices, while [8] returns all marked ones. This is one place where we can save a logarithmic factor, which requires more careful analysis.

We bound the facility and connection cost of the solution S given by Algorithm 1 respectively. Indeed, for the facility cost, we prove some stronger statement. Define $V^\circ = \{u \in V_t : N_u \geq 1\}$ be the set of vertices u with at least 1 client in T_u . We prove

Claim 2. $S \subseteq \text{min-set}(V^\circ \cap M)$.

The stronger statement we prove about the facility cost of the solution S is the following:

Lemma 1. $|\text{min-set}(V^\circ \cap M)| \cdot f \leq (1 + 1/\epsilon)\text{opt}$.

Notice that Claim 2 and Lemma 1 imply that $|S| \cdot f \leq q(1 + 1/\epsilon)\text{opt}$.

Now we switch gear to consider the connection cost of the solution S and prove:

Lemma 2. The connection cost of S given by the base algorithm is at most $O(1)\text{opt}$.

4.3 Guaranteeing ϵ -DP by Adding Noises

In this section, we describe the final algorithm (Algorithm 2) that achieves ϵ -DP without sacrificing the order of the approximation ratio. Recall that $\eta = \sqrt{\lambda}$.

Algorithm 2 DP-UFL-tree(ϵ)

- 1: for every $v \in V_T$ with $\ell(v) < L'$, define $\tilde{N}_v := N_v + \text{Lap}\left(\frac{f}{c\eta^{L'+\ell(v)}}\right)$, where $c = \frac{\eta-1}{\eta}$.
 - 2: Let $M \leftarrow \left\{ v \in V_T : \ell(v) \geq L' \text{ or } \tilde{N}_v \cdot \lambda^{\ell(v)} \geq f \right\}$ be the set of marked vertices
 - 3: $R \leftarrow \text{min-set}(M)$
 - 4: **return** R but only open $S \subseteq R$ using the closest-facility rule.
-

We give some intuitions on how we choose the noises in Step 1 of the Algorithm. Let us travel through the tree from level L' down to level 0. Then the Laplacian parameter, which corresponds to the magnitude of the Laplacian noise, goes up by factors of η . This scaling factor is carefully chosen to guarantee two properties. First the noise should go up exponentially so that the DP parameter only depends on the noise on the highest level, i.e, level L' . Second, η is smaller than the distance scaling factor $\lambda = \eta^2$. Though the noises are getting bigger as we travel down the tree, their effects are getting smaller since they do not grow fast enough. Then essentially, the effect of the noises is only on levels near L' .

Lemma 3. Algorithm 2 satisfies ϵ -DP property.

4.4 Increase of cost due to the noises

We shall analyze how the noise affects the facility and connection costs. Let M^0, R^0 and S^0 (resp. M^1, R^1 and S^1) be the M, R and S generated by Algorithm 1 (resp. Algorithm 2). In the proof, we shall also consider running Algorithm 1 with input vector being $2\vec{N}$ instead of \vec{N} . Let M'^0, R'^0

and S^0 be the M , R and S generated by Algorithm 1 when the input vector is $2\vec{N}$. Notice that the optimum solution for input vector $2\vec{N}$ is at most 2opt . Thus, Lemma 1 implies $|S^0| \cdot f = O(1/\epsilon)\text{opt}$. Notice that M^0, R^0, S^0, M^0, R^0 and S^0 are deterministic while M^1, R^1 and S^1 are randomized.

The lemmas we shall prove is the following:

Lemma 4. $\mathbb{E}[|S^1| \cdot f] \leq O(1/\epsilon) \cdot \text{opt}$.

Lemma 5. The expected connection cost of the solution S^1 is $O(1)$ times that of S^0 .

Thus, combining the two lemmas, we have that the expected cost of the solution S^1 is at most $O(1/\epsilon)\text{opt}$, finishing the proof of Theorem 2. Indeed, we only lose an $O(1)$ -factor for the connection cost as both factors in Lemma 1 and 5 are $O(1)$. We then prove the two lemmas separately.

4.4.1 Increase of facility costs due to the noise

In this section, we prove Lemma 4. A technical lemma we can prove is the following:

Claim 3. Let $M \subseteq V_T$ and $M' = M \cup \{v\}$ for some $v \in V_T \setminus M$, then exactly one of following three cases happens.

- (3a) $\text{min-set}(M') = \text{min-set}(M)$.
- (3b) $\text{min-set}(M') = \text{min-set}(M) \uplus \{v\}$.
- (3c) $\text{min-set}(M') = \text{min-set}(M) \setminus \{u\} \cup \{v\}$, where $u \in \text{min-set}(M)$, $v \notin \text{min-set}(M)$ and v is a descendant of u .

Proof of Lemma 4. Recall that V° is the set of vertices u with $N_u \geq 1$. We first focus on open facilities in V° in S^1 . Claim 3 implies that adding one new element to M will increase $|\text{min-set}(M)|$ by at most 1. Thus, we have

$$\begin{aligned} & |\text{min-set}(M^1 \cap V^\circ)| - |\text{min-set}(M^0 \cap V^\circ)| \leq |(M^1 \cap V^\circ) \setminus (M^0 \cap V^\circ)| \\ & = \left| \left\{ u \in V^\circ : \ell(u) < L', 2N_u < \frac{f}{\lambda^{\ell(u)}} \leq \tilde{N}_u \right\} \right|. \end{aligned}$$

We now bound the expectation of the above quantity. Let U^* be the set of vertices $u \in V^\circ$ with $\ell(u) < L'$ and $N_u < \frac{f}{2\lambda^{\ell(u)}}$. Then for every $u \in U^*$, we have

$$\begin{aligned} \Pr[u \in M^1] &= \Pr \left[N_v + \text{Lap} \left(\frac{f}{c\eta^{L'+\ell(u)}} \right) \geq \frac{f}{\lambda^{\ell(u)}} \right] \\ &\leq \frac{1}{2} \exp \left(-\frac{f/(2\lambda^{\ell(u)})}{f/(c\eta^{L'+\ell(u)})} \right) = \frac{1}{2} \exp \left(-\frac{c\eta^{L'-\ell(u)}}{2} \right). \end{aligned} \quad (2)$$

We bound f times the sum of (2), over all $u \in U^*$. Notice that every $u \in V^\circ$ has $N_u \geq 1$. So we have $B_u = \min \{f, N_u \lambda^{\ell(u)}\} \geq \lambda^{\ell(u)}$ for every u we are interested. Then,

$$f \leq \frac{1}{\epsilon} \cdot \epsilon f \cdot \frac{B_u}{\lambda^{\ell(u)}} \leq \frac{1}{\epsilon} \cdot \lambda^{L'} \cdot \frac{B_u}{\lambda^{\ell(u)}} = \frac{B_u}{\epsilon} \cdot \eta^{2(L'-\ell(u))}. \quad (3)$$

The last inequality comes from $\epsilon f \leq \lambda^{L'}$. The equality used that $\lambda = \eta^2$.

We group the u 's according to $\ell(u)$. For each level $\ell \in [0, L' - 1]$, we have

$$\frac{f}{2} \sum_{u \in U^*: \ell(u)=\ell} \exp \left(-\frac{c\eta^{L'-\ell(u)}}{2} \right) \leq \frac{1}{2\epsilon} \eta^{2(L'-\ell)} \exp \left(-\frac{c\eta^{L'-\ell}}{2} \right) \sum_{u \text{ as before}} B_u \leq \frac{c_\ell}{2\epsilon} \text{opt},$$

where we defined $x_\ell = \eta^{L'-\ell(u)}$ and $c_\ell = x_\ell^2 \exp(-\frac{cx_\ell}{2})$. The last inequality used Claim 1, which holds since all u 's in the summation are at the same level.

Taking the sum over all ℓ from 0 to L' , we obtain

$$\sum_{u \in U^*} \Pr[u \in M^1] \leq \frac{\text{opt}}{2\epsilon} \cdot \sum_{\ell=0}^{L'-1} c_\ell = \frac{\text{opt}}{2\epsilon} \cdot \sum_{\ell=0}^{L'-1} x_\ell^2 \exp(-\frac{cx_\ell}{2}).$$

Notice that $\{x_\ell : \ell \in [0, L' - 1]\}$ is exactly $\{\eta, \eta^2, \dots, \eta^{L'}\}$. It is easy to see summation is bounded by a constant for any constant c . Thus, the above quantity is at most $O(1/\epsilon)\text{opt}$. Therefore, we proved $\mathbb{E}[|\text{min-set}(M^1 \cap V^\circ)| - |\text{min-set}(M^{l_0} \cap V^\circ)|] \leq O(1/\epsilon) \cdot \text{opt}$.

Notice that Lemma 1 says that $|\text{min-set}(M^{l_0} \cap V^\circ)| \leq O(1/\epsilon)\text{opt}$. Thus $\mathbb{E}[|\text{min-set}(M^1 \cap V^\circ)|] \leq O(1/\epsilon)\text{opt}$.

Then we take vertices outside V° into consideration. Let $U = \text{min-set}(M^1 \cap V^\circ)$. Then $S^1 \subseteq R^1 = \text{min-set}(U \cup (V_T \setminus V^\circ))$. To bound the facility cost of S^1 , we start with the set $U' = U$ and add vertices in $V_T \setminus V^\circ$ (these are vertices u with $N_u = 0$) to U' one by one and see how this changes $\text{min-set}(U')$. By Claim 3, adding a vertex $N_v = 0$ to U' will either not change $\text{min-set}(U')$, or add v to $\text{min-set}(U')$, or replace an ancestor of v with v . In all the cases, the set $\text{min-set}(U') \cap V^\circ$ can only shrink. Thus, we have $R^1 \cap V^\circ \subseteq \text{min-set}(U) = \text{min-set}(M^1 \cap V^\circ)$. We have $E[|R^1 \cap V| \cdot f] \leq O(1/\epsilon) \cdot \text{opt}$.

Thus, it suffices to bound the expectation of $|S \setminus V^\circ| \cdot f$. Focus on some $u \in V_T$ with $N_u = 0$. Notice that $u \notin S$ if $\ell(u) \geq L'$. So, we assume $\ell(u) < L'$. In this case there is some leaf $v \in V$ with $N_v > 0$ such that u is the closest point in R to v . So v is not an descendant of u . Let u' be the ancestor of v that is at the same level as u and define $\pi(u) = u'$. Then $\ell(\pi(u)) = \ell(u)$. Moreover, u is also the closest point in R to u' , implying that π is an injection. For every u , we can bound f as in (3), but with B_u replaced by $B_{\pi(u)}$. Then the above analysis still works since we have $\sum_{u: N_u=0, \ell(u)=\ell} B_{\pi(u)} \leq \text{opt}$ for every $\ell \in [0, L' - 1]$ by Claim 1. \square

4.4.2 Increase of connection cost due to the noise

Now we switch gear to consider the change of connection cost due to the noise.

Proof of Lemma 5. Focus on a vertex v at level ℓ and suppose $v \in S^0$ and some clients are connected to v in the solution produced by Algorithm 2. So, we have $N_v \geq \frac{f}{\lambda^\ell}$. Let the ancestor of v (including v itself) be $v_0 = v, v_1, v_2, \dots$ from the bottom to the top. Then the probability that $v_0 \notin M^0$ is at most $1/2$ and in that case the connection cost increases by a factor of λ . The probability that $v_0, v_1 \notin M^0$ is at most $1/4$, and in that case the cost increases by a factor of λ^2 and so on. As a result, the expected scaling factor for the connection cost due to the noise is at most

$$\sum_{i=0}^{\infty} \frac{1}{2^i} \cdot \lambda^i = \sum_{i=1}^{\infty} \left(\frac{\lambda}{2}\right)^i = O(1).$$

Thus, the connection cost of the solution S^1 is at most a constant times that of S^0 . This is the place where we require $\lambda < 2$. \square

5 Lower Bound of UFL for HST Metric

In this section, we prove an $\Omega(1/\sqrt{\epsilon})$ lower bound on the approximation ratio of any algorithm for UFL in the super-set setting under the ϵ -DP model. The metric we are using is the uniform star-metric: the shortest-path metric of a star where all edges have the same length. We call the number of edges in the star its size and the length of these edges its *radius*. By splitting edges, we can easily see that the metric is a λ -HST metric for a $\lambda > 1$, if the radius is $\frac{\lambda^L}{\lambda-1}$ for some integer L .

The main theorem we are going to prove is the following:

Theorem 4. There is a constant $c > 0$ such that the following holds. For any small enough $\epsilon < 1$, $f > 0$ and sufficiently large integer n that depends on ϵ , there exists a set of UFL instances $\{(V, d, f, \vec{N})\}_{\vec{N}}$, where (V, d) is the uniform-star metric of size n and radius $\sqrt{\epsilon}f$, and every instance in the set has $n \leq |\vec{N}|_1 \leq n/\epsilon$, such that the following holds: no ϵ -DP algorithm under the super-set setting can achieve $c\frac{1}{\sqrt{\epsilon}}$ -approximation for all the instances in the set.

Throughout the proof, we let $m = 1/\epsilon$ and we assume m is an integer. We prove Theorem 4 in two steps, first we show the lower bound on an instance with a 2-point metric, but non-uniform facility costs. Then we make the facility costs uniform by combining multiple copies of the 2-point metric into a star metric.

Consider the instance shown in Figure 1a where $V = \{a, b\}$ and $d(a, b) = \sqrt{\epsilon}f$. The facility costs for a and b are respectively f and 0 . Thus, we can assume the facility b is always open. All the clients are at a , and the number N of clients is promised to be an integer between 1 and m . We show that for this instance, no ϵ -DP algorithm in the super-set output setting can distinguish between the case where $N = 1$ and that $N = m$ with constant probability; this will establish the $\Omega(\sqrt{m})$ lower bound.

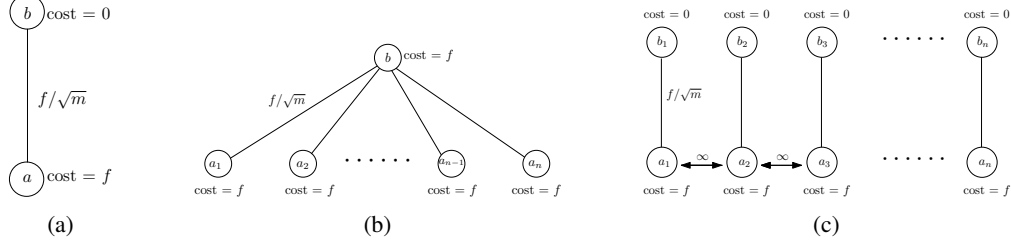


Figure 1: Instance for the lower bound.

Obliviously, there are only 2 solutions for any instance in the setting: either we open a , or we do not. Since we are promised there is at least 1 client, the central curator has to reveal whether we open a or not, even in the super-set output setting: If we do not open a , then we should not include a in the returned set R since otherwise the client will think it is connected to a ; if we open a , then we need to include it in the returned set R since all the clients need to be connected to a .

Let D_i be the scenario where we have $N = i$ clients at a , where $i \in [m]$. Then the cost of the two solutions for the two scenarios D_1 and D_m are listed in the following table:

	not open a	open a
D_1	f/\sqrt{m}	f
D_m	$\sqrt{m}f$	f

Thus, if the data set is D_1 , we should not open a ; if we opened, we'll lose a factor of \sqrt{m} . If the data set is D_m , then we should open a ; if we did not open, then we also lose a factor of \sqrt{m} .

Now consider any ϵ -DP algorithm \mathcal{A} . Assume towards the contradiction that \mathcal{A} achieves $0.2\sqrt{m}$ approximation ratio. Then, under the data set D_1 , \mathcal{A} should choose not to open a with probability at least 0.8 . By the ϵ -DP property, under the data set D_m , \mathcal{A} shall choose not to open a with probability at least $0.8e^{-(m-1)\epsilon} > 0.8/e \geq 0.2$. Then under the data set D_m , the approximation ratio of \mathcal{A} is more than $0.2\sqrt{m}$, leading to a contradiction.

Indeed, later we need an average version of the lower bound as follows:

$$\frac{\sqrt{m}}{\sqrt{m}+1} \mathbb{E} \text{cost}(\mathcal{A}(1); 1) + \frac{1}{\sqrt{m}+1} \mathbb{E} \text{cost}(\mathcal{A}(m); m) \geq cf, \quad (4)$$

where c is an absolute constant, $\mathcal{A}(N)$ is the solution output by the algorithm \mathcal{A} when there are N clients at a , and $\text{cost}(\mathcal{A}(N); N)$ is the cost of the solution under the input N . Our argument above showed that either $\mathbb{E} \text{cost}(\mathcal{A}(1); 1) \geq \Omega(0.2\sqrt{m}) \cdot f/\sqrt{m} = 0.2f$, or $\mathbb{E} \text{cost}(\mathcal{A}(N); N) \geq 0.2\sqrt{m} \cdot f = 0.2\sqrt{m}f$. In either case, the left side of (4) is at least $\frac{0.2\sqrt{m}f}{\sqrt{m}+1} \geq cf$ if c is small.

The above proof almost proved Theorem 4 except that we need to place a free open facility at location b . To make the facility costs uniform, we can make multiple copies of the locations a , while only keeping one location b ; this is exactly a star metric (see Figure 1b). The costs for all facilities are f . However, since there are so many copies of a , the cost of f for opening a facility at b is so small and thus can be ignored. Then, the instance essentially becomes many separate copies of the 2-point instances we described (see Figure 1c).

However, proving that the ‘‘parallel repetition’’ instance in Figure 1c has the same lower bound as the original two-point instance is not so straightforward. Intuitively, we can imagine that the central curator should treat all copies independently: the input data for one copy should not affect the decisions we make for the other copies. However, it is tricky to prove this. Instead, we prove Theorem 4 directly by defining a distribution over all possible instances and argue that an ϵ -DP algorithm must be bad on average. Due to the page limit, the detailed analysis is left to the supplementary material.

References

- [1] Vijay Arya, Naveen Garg, Rohit Khandekar, Adam Meyerson, Kamesh Munagala, and Vinayaka Pandit. Local search heuristics for k-median and facility location problems. *SIAM Journal on computing*, 33(3):544–562, 2004.
- [2] Maria-Florina Balcan, Travis Dick, Yingyu Liang, Wenlong Mou, and Hongyang Zhang. Differentially private clustering in high-dimensional euclidean spaces. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 322–331. JMLR. org, 2017.
- [3] Adrian Rivera Cardoso and Rachel Cummings. Differentially private online submodular minimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1650–1658, 2019.
- [4] Moses Charikar and Sudipto Guha. Improved combinatorial algorithms for the facility location and k-median problems. In *40th Annual Symposium on Foundations of Computer Science*, pages 378–388. IEEE, 1999.
- [5] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284. Springer, 2006.
- [6] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *Journal of Computer and System Sciences*, 69(3):485–497, 2004.
- [7] Dan Feldman, Amos Fiat, Haim Kaplan, and Kobbi Nissim. Private coresets. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 361–370. ACM, 2009.
- [8] Anupam Gupta, Katrina Ligett, Frank McSherry, Aaron Roth, and Kunal Talwar. Differentially private combinatorial optimization. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 1106–1125. Society for Industrial and Applied Mathematics, 2010.
- [9] Justin Hsu, Zhiyi Huang, Aaron Roth, Tim Roughgarden, and Zhiwei Steven Wu. Private matchings and allocations. *SIAM Journal on Computing*, 45(6):1953–1984, 2016.
- [10] Justin Hsu, Zhiyi Huang, Aaron Roth, and Zhiwei Steven Wu. Jointly private convex programming. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pages 580–599. Society for Industrial and Applied Mathematics, 2016.
- [11] Zhiyi Huang and Xue Zhu. Near optimal jointly private packing algorithms via dual multiplicative weight update. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 343–357. Society for Industrial and Applied Mathematics, 2018.
- [12] Zhiyi Huang and Xue Zhu. Scalable and jointly differentially private packing. *arXiv preprint arXiv:1905.00767*, 2019.
- [13] Kamal Jain and Vijay V Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *Journal of the ACM (JACM)*, 48(2):274–296, 2001.
- [14] Michael Kearns, Mallesh Pai, Aaron Roth, and Jonathan Ullman. Mechanism design in large games: Incentives and privacy. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pages 403–410. ACM, 2014.
- [15] Shi Li. A 1.488 approximation algorithm for the uncapacitated facility location problem. In *International Colloquium on Automata, Languages, and Programming*, pages 77–88. Springer, 2011.
- [16] Shi Li. On facility location with general lower bounds. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2279–2290. Society for Industrial and Applied Mathematics, 2019.

- [17] Marko Mitrovic, Mark Bun, Andreas Krause, and Amin Karbasi. Differentially private submodular maximization: data summarization in disguise. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2478–2487. JMLR. org, 2017.
- [18] Kobbi Nissim, Rann Smorodinsky, and Moshe Tennenholtz. Approximately optimal mechanism design via differential privacy. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 203–213, 2012.
- [19] David B Shmoys and KI Aardal. *Approximation algorithms for facility location problems*, volume 1997. Utrecht University: Information and Computing Sciences, 1997.
- [20] Jean-Michel Thizy. A facility location problem with aggregate capacity. *INFOR: Information Systems and Operational Research*, 32(1):1–18, 1994.

article

[final]neurips2019

[utf8]inputenc [T1]fontenc hyperref url booktabs amsfonts nicefrac microtype amsfonts nicefrac microtype times adjustbox tabularx amsmath amssymb amsthm blindtext enumitem multirow graphicx subcaption algorithmalgpseudocode Assumption Theorem Definition Corollary Lemma Remark Claim Fact color

*E E

*U U

A Connection of our setting to the Joint Differential Privacy model

The *Joint Differential Privacy* model was initially proposed in [Kears-Pai-Roth-Ullman, ITCS 2014]. In the model, every client gets its own output from the central curator and the algorithm is ϵ -joint differentially private (JDP) if for every two datasets D, D' with $D' = D \uplus \{j\}$, the joint distribution of the outputs for all clients *except* j under the data D is not much different from that under the dataset D' (using a definition similar to that of the ϵ -Differential Privacy). In other words, j 's own output should not be considered when we talk about the privacy for j .

The super-set output setting in the ϵ -DP model we considered for UFL is closely related to the JDP-model. In order for the JDP model to be meaningful, one needs to define the problem in such a way that the algorithm needs to give an output for each client, which contains important information for the client. In the UFL problem, this information can be which facility the client should be connected to. If we define UFL in this way, i.e. every client only needs to know its own connecting facility, then our ϵ -DP algorithm in the super-set output setting implies an ϵ -JDP algorithm: Instead of outputting the superset R to everyone, the central curator can simply output to each j the facility that j is connecting to.

B Reducing general metrics to HST metrics: Proof of Theorem 1

This section is dedicated to the proof of Theorem 1. The FRT tree decomposition result states that any n -point metric can be embedded into a distribution of $O(1)$ -HST metrics:

Theorem V. There is an efficient randomized algorithm that, given a constant $\lambda > 1$, a metric (V, d) with $n := |V|$, minimum non-zero distance at least 1 and diameter $\Delta := \max_{u,v \in V} d(u, v)$, outputs a λ -HST T whose leaves are exactly V satisfying the following conditions:

(Va) With probability 1, for every $u, v \in V$, we have $d_T(u, v) \geq d(u, v)$.

(Vb) For every $u, v \in V$, we have

$$\mathbb{E} d_T(u, v) \leq O(\log n) \cdot d(u, v),$$

where the expectation is over all randomness of the algorithm.

(Va) says that the metric d_T restricted to V is always *non-contracting* compared to the metric d . On the other hand (Vb) says that d_T “expands” only by a factor of $O(\log n)$ in expectation. Theorem V allows us to reduce many combinatorial problems with min-sum objective on general metrics to those on metrics induced by HSTs. We use the UFL problem as an in this proof.

Let $\lambda > 1$ be a small enough constant as in Theorem 1. Given the UFL instance (V, d, f, \vec{N}) , our algorithm shall randomly generate a λ -HST T for the input metric (V, d) using the algorithm in Theorem V. Then we solve the instance $(V, d_T|_V, f, \vec{N})$ using the algorithm \mathcal{A} to obtain a set S of open facilities.⁴ One show that the final approximation ratio we obtain for the original instance (V, d, f, \vec{N}) will be $O(\log n)$ times that for the instance $(V, d_T|_V, f, \vec{N})$; moreover the ϵ -DP property carries over in the reduction.

⁴Recall that in the super-set output setting, S is not the returned set, but the set of facilities that are connected by at least 1 client. One small caveat is that for the original instance (V, d, f, \vec{N}) , given a set R of facilities returned by the algorithm, we should use the tree metric d_T to decide how to connect the clients, instead of the original metric d . Thus, along with the set R , the algorithm should also return the HST T .

Claim IV. The expected cost of S to the original instance (V, d, f, \vec{N}) is at most $O(\alpha_{\text{tree}} \cdot \log n)$ times that of the optimum solution of (V, d, f, \vec{N}) .

Proof. Let $S^* \subseteq V$ be the optimum solution to the instance (V, d, f, \vec{N}) . Then, we have

$$\frac{\mathbb{E}}{T} \text{cost}_{d_T}(S^*) \leq O(\log n) \text{cost}_d(S^*).$$

Above, $\text{cost}_{d_T}(S^*)$ is the cost of the solution S^* w.r.t instance $(V, d_T|_V, f, \vec{N})$ and $\text{cost}_d(S^*)$ is the cost of the solution S^* w.r.t (V, d, f, \vec{N}) . The expectation is over the randomness of T . The equation simply follows from property (Vb).

If the algorithm for the instance $(V, d_T|_V, f, \vec{N})$ is an α_{tree} -approximation, then we have

$$\mathbb{E}[\text{cost}_{d_T}(S)|T] \leq \alpha_{\text{tree}} \cdot \text{cost}_{d_T}(S^*),$$

where S is the solution output by the algorithm when the chosen tree is T . This holds since the optimum cost to the instance $(V, d_T|_V, f, \vec{N})$ is at most $\text{cost}_{d_T}(S^*)$. Then, using property (Va), i.e., the metrics are non-contracting, we have $\text{cost}_d(S) \leq \text{cost}_{d_T}(S)$. Thus, we have

$$\mathbb{E}[\text{cost}_d(S)|T] \leq \alpha_{\text{tree}} \cdot \text{cost}_{d_T}(S^*).$$

Taking the expectation over the randomness of T , we have

$$\frac{\mathbb{E}}{T} \text{cost}_d(S) = \frac{\mathbb{E}}{T} [\mathbb{E}[\text{cost}_d(S)|T]] \leq \alpha_{\text{tree}} \cdot \frac{\mathbb{E}}{T} \text{cost}_{d_T}(S^*) \leq O(\alpha_{\text{tree}} \cdot \log n) \text{cost}_d(S^*).$$

□

For the ϵ -DP property of the algorithm, it suffices to observe that the random variable T is independent of \vec{N} , and for any fixed T , the algorithm \mathcal{A} on T is ϵ -DP. This finishes the proof of Theorem 1.

C Missing Proofs from Section 4

Proof of Claim 1. Fix the optimum solution for the instance (V, d_T, \vec{N}) . We define the cost inside a sub-tree T_v to be the total cost of open facilities in T_v , and the connection cost of clients in T_v in the optimum solution. Clearly, opt is at least the sum of costs inside T_v over all $v \in V'$ since the $\{T_v\}_{v \in V'}$ are disjoint. On the other hand, the cost inside T_v is at least $B_v = \min\{f, N_v \lambda^{\ell(v)}\}$: If T_v contains some open facility, then the cost is at least f ; otherwise, all clients in T_v has to be connected to outside T_v , incurring a cost of at least $N_v \lambda^{\ell(v)}$. The claim then follows. □

Proof of Claim 2. We prove that any $u \in S$ also has $u \in \text{min-set}(V^\circ \cap M)$. Notice that any u with $\ell(u) > L'$ will not be in R (as it will not be minimal-marked) and thus it will not be in S . Any u with $\ell(u) < L'$ and $N_u = 0$ will not be marked and thus will not be in S .

Now consider the case $\ell(u) \leq L'$ and $N_u \geq 1$. If $u \in S \subseteq R$ then u is minimal-marked in M . It must also be minimal-marked in $M \cap V^\circ$ since all vertices in $V_T \setminus V^\circ$ below level L' , in particular, vertices in $V_T \setminus V^\circ$ that are descendants of u , are not marked. So $u \in \text{min-set}(M \cap V^\circ)$.

Thus, it remains to focus on a vertex u with $\ell(u) = L'$ and $N_u = 0$. Consider any leaf v with $N_v > 0$ and the level- L' ancestor u' of v ; so $u' \neq u$ since $N_u = 0$. Then $u' \in M$ and there will be some $u'' \in T_{u'}$ such that $u'' \in R$. We have $d(v, u'') < d(v, u)$ and thus clients in v will not be connected to u . This holds for any v with $N_v > 0$. So, $u \notin S$. The finishes the proof of the claim. □

Proof of Lemma 1. We break the set of u 's in $\text{min-set}(V^\circ \cap M)$ into two cases and bound the total cost for each case separately.

- (a) $\ell(u) = L'$. By the definition of V° , we have $N_u \geq 1$. Then, $f \leq \frac{1}{\epsilon} \lambda^{L'} \leq \frac{1}{\epsilon} N_u \lambda^{\ell(u)}$ and thus $f \leq \frac{1}{\epsilon} B_u$ by the definitions of L' and B_u . So, we have the total facility cost of all u 's in this case is at most $\frac{1}{\epsilon} \sum_{u \text{ in the case (a)}} B_u \leq \frac{1}{\epsilon} \cdot \text{opt}$. The inequality is by Claim 1, which holds as all u 's in the summation are at the same level.

- (b) $\ell(u) < L'$. u must minimal-marked, i.e., $u \in R$. Then we have $f \leq N_u \lambda^{\ell(u)}$ and thus $f \leq B_u$. The total cost in this case is at most $\sum_{u \in R} B_u \leq \text{opt}$ by Claim 1, which holds since R does not contain an ancestor-descendant pair.

Thus, we have proved $F \leq (1 + 1/\epsilon)\text{opt}$. Combining the upper bounds for both C and F gives the lemma. \square

Proof of Lemma 2. Notice that reducing R to S will not increase the connection cost of any client since we are using the closest-facility rule. Thus, we can pretend the set of open facilities is R instead of S . Let us focus the clients located at any leaf $v \in V$. Let u be the level- L' ancestor of v ; notice that u is marked. If v is marked, then $v \in R$ and thus the connection cost of any client at v is 0. So we can assume v is unmarked. Then there is exactly one maximal-unmarked vertex $u' \neq u$ in the u - v path in T . Let u'' be the parent of u' . Then u'' is marked and some vertex in $T_{u''}$ will be in R . So the connection cost of a client at v is at most $\frac{2}{\lambda-1} \lambda^{\ell(u'')} = \frac{2\lambda}{\lambda-1} \cdot \lambda^{\ell(u')}$. The total connection cost of clients is at most

$$\frac{2\lambda}{\lambda-1} \sum_{u' \in V_T \text{ is maximal-unmarked}} N_{u'} \cdot \lambda^{\ell(u')} = \frac{2\lambda}{\lambda-1} \sum_{u' \text{ as before}} B_{u'} \leq \frac{2\lambda}{\lambda-1} \cdot \text{opt}.$$

For every u' in the summation, we have $N_{u'} \lambda^{\ell(u')} < f$ and thus $B_{u'} = N_{u'} \lambda^{\ell(u')}$. So the equality in the above sequence holds. The inequality follows from Claim 1, as the set of maximal-unmarked vertices does not contain an ancestor-descendant pair. Thus, we proved the lemma. \square

Proof of Lemma 3. We focus on two datasets $\vec{N}, \vec{N}' \in \mathbb{Z}_{\geq 0}^V$ satisfying $|\vec{N}' - \vec{N}|_1 = 1$. Let v be the unique leaf with $|N_v - N'_v| = 1$. We show the ϵ -differential privacy between the two distributions of M generated by Algorithm 2 for the two datasets. Since the returned set R is completely decided by M , this is sufficient to establish the ϵ -DP property. For two different vertices $u, u' \in V_T$, the event $u \in M$ is independent of the event $v \in M$, since the former only depends on the noise added to N_u and the later only depends that added to $N_{u'}$. Thus, by the composition of differentially private mechanisms, we can focus on the sub-algorithm that only outputs whether $u \in M$, for each $u \in V_T$.

For simplicity, for every $u \in V_T$, let a_u and a'_u respectively indicate if $u \in M$ under the datasets \vec{N} and \vec{N}' . If $\ell(u) \geq L'$ then $a_u = a'_u = 1$ always holds. Also, a_u and a'_u have the same distribution if u is not an ancestor of v , since $\vec{N}_u = \vec{N}'_u$. That is the sub-algorithm for u is 0-DP between \vec{N} and \vec{N}' .

So we can focus on an ancestor u of v with $\ell(u) < L'$. For this u we have $|N_u - N'_u| = 1$. Due to the property of the Laplacian distribution, the sub-algorithm for this u is $\frac{c\eta^{L'+\ell(v)}}{f}$ -differentially private between \vec{N} and \vec{N}' . Summing the privacy over all such vertices gives the total privacy as

$$c \sum_{\ell=0}^{L'-1} \frac{\eta^{L'+\ell(v)}}{f} = \frac{c\eta^{L'}}{f} \cdot \frac{\eta^{L'} - 1}{\eta - 1} \leq \frac{c}{f(\eta - 1)} \cdot \lambda^{L'} \leq \frac{c}{f(\eta - 1)} \cdot \lambda \epsilon f = \frac{c\eta\epsilon}{\eta - 1} = \epsilon,$$

by the definition of c . \square

Proof of Claim 3. Notice that $\text{min-set}(M') = \text{min-set}(\text{min-set}(M) \cup \{v\})$, and $\text{min-set}(M)$ does not contain an ancestor-descendant pair. If v is an ancestor of some vertex in $\text{min-set}(M)$, Case (3a) happens. If v is a descendant of some vertex in $\text{min-set}(M)$, then Case (3c) happens. Otherwise, we have case (3b). \square

D Lower Bound on Approximation Ratio for ϵ -DP Algorithm: Making Facility Costs Uniform

In this section, we formally describe the star instance we shall construct for proving Theorem 4. The star is depicted in Figure 1b: the set of nodes are $V = \{b, a_1, a_2, \dots, a_n\}$ where b is the center and a_i 's are the leaves. The distance between b and a_i for $i \in [n]$ is f/\sqrt{m} . The facility cost of each node equals to f .

We shall make sure that in every location a_i , we have at least 1 client. Then, the optimum solution to the instance has cost at least $n \cdot f/\sqrt{m}$. If our n is at least, say $\sqrt{m} = 1/\sqrt{\epsilon}$, then the cost is at least f . So, opening the facility at b will only cost f and thus only lose an additive factor of 1 in the approximation ratio. This does not affect our analysis since we are interested in whether there is a $c\sqrt{m}$ -approximation or not. Thus, without loss of generality we can assume the facility at b is open for free. With this assumption, we can see that the instance in Figure 1b is equivalent to instance in Figure 1c : it can be seen as n copies of the instance in Figure 1a, that is for each $i \in [n]$, the facility cost at b_i is 0, while it is f at a_i . The distance between b_i and a_i is f/\sqrt{m} . Moreover, the distances between the copies are ∞ . In the following we will focus on instance in Figure 1c.

We can use a vector $\vec{N} \in \mathbb{Z}_{\geq 0}^n$ to denote an input vector, where N_i is the number of clients at location a_i . We shall only be interested in the datasets $\vec{N} \in \{1, m\}^n$, i.e, every location has either 1 or m clients. We define a distribution \mathcal{P} over the set of vectors as follows: for every $i \in [n]$, let $N_i = 1$ with probability $\frac{\sqrt{m}}{\sqrt{m+1}}$ and $N_i = m$ with probability $\frac{1}{\sqrt{m+1}}$; the choices for all $i \in [n]$ are made independently.

We consider the term of $\mathbb{E}_{\vec{N} \sim \mathcal{P}}[\text{cost}(S^*(\vec{N}); \vec{N})]$, where $S^*(\vec{N})$ is the optimal solution for the dataset \vec{N} . Then, for the optimum solution is easy to define: if $N_i = 1$, we do not open a_i and if $N_i = m$, we open a_i . The expected cost of the optimum solution over all vector \vec{N} is

$$\begin{aligned} \mathbb{E}_{\vec{N} \sim \mathcal{P}}[\text{cost}(S^*(\vec{N}); \vec{N})] &= \sum_{\vec{N} \in \{1, m\}^n} P_{\vec{N}}[\text{cost}(S^*(\vec{N}); \vec{N})] = \sum_{\vec{N} \in \{1, m\}^n} P_{\vec{N}} \sum_{i=1}^n \text{opt}(N_i) \\ &= \sum_{i=1}^n \sum_{\vec{N}, \vec{N}' \in \{1, m\}^n, \vec{N}, \vec{N}' \text{ differ at } i, N_i=1, N'_i=m} [P_{\vec{N}} \cdot \text{opt}(1) + P_{\vec{N}'} \cdot \text{opt}(m)] \\ &= \sum_{i, \vec{N}, \vec{N}' \text{ as before}} \frac{2P_{\vec{N}}f}{\sqrt{m}}. \end{aligned} \tag{v}$$

Above $\text{opt}(1) = f/\sqrt{m}$ and $\text{opt}(m) = f$ are the optimum cost for the two-point instance when there are 1 and m clients respectively. $P_{\vec{N}}$ is the probability of \vec{N} according to \mathcal{P} . The second equality in (v) is due to the structure of instance of Figure 1c and thus $\text{cost}(S^*(\vec{N}); \vec{N}) = \sum_{i=1}^n \text{opt}(N_i)$. Notice that we could have a cleaner form for the expected cost; however we use the above form for the purpose of easy comparison.

Now we consider any ϵ -DP algorithm \mathcal{A} . Due to the structure of Figure 1c, if we denote the term $\text{cost}_i(\mathcal{A}(N_i); N_i)$ as the cost of $\mathcal{A}(\vec{N})$ given \vec{N} incurred by clients at a_i , the expected cost of \mathcal{A} can be written as

$$\begin{aligned} \mathbb{E}_{\vec{N} \sim \mathcal{P}, \mathcal{A}}[\text{cost}(\mathcal{A}(\vec{N}); \vec{N})] \\ = \sum_{i=1}^n \sum_{\vec{N}, \vec{N}' \in \{1, m\}^n, \vec{N}, \vec{N}' \text{ differ at } i, N_i=1, N'_i=m} [P_{\vec{N}} \mathbb{E} \text{cost}_i(\mathcal{A}(\vec{N}); \vec{N}) + P_{\vec{N}'} \mathbb{E} \text{cost}_i(\mathcal{A}(\vec{N}'); \vec{N}')]. \end{aligned}$$

From the analysis of the two-point metric, we already proved that for any such pair (\vec{N}, \vec{N}') in the summation, we have

$$P_{\vec{N}} \mathbb{E} \text{cost}_i(\vec{N}; \mathcal{A}(\vec{N})) + P_{\vec{N}'} \mathbb{E} \text{cost}_i(\vec{N}'; \mathcal{A}(\vec{N}')) \geq (P_{\vec{N}} + P_{\vec{N}'})cf \geq P_{\vec{N}}cf. \tag{vi}$$

Indeed, if the above property does not hold, we can just use our algorithm \mathcal{A} as a black box to solve the two-point instance: For every $j \neq i$, we just pretend we have $N_j = N'_j$ clients at a_j ; the number of clients at a_i is exactly the number of clients in a in the two point solution; Run the algorithm \mathcal{A} on the instance. Then the negation of (vi) will imply the negation of (4). Thus (vi) must hold.

Thus we have

$$\mathbb{E}_{\vec{N} \sim \mathcal{P}}[\text{cost}(\mathcal{A}(\vec{N}); \vec{N})] \geq \sum_{i, \vec{N}, \vec{N}'} cP_{\vec{N}}f. \tag{vii}$$

Comparing the inequality with (v) gives that

$$\mathbb{E}_{\vec{N} \sim \mathcal{P}}[\text{cost}(\mathcal{A}(\vec{N}); \vec{N})] \geq (c\sqrt{m}/2)\mathbb{E}_{\vec{N} \sim \mathcal{P}}[\text{cost}(S^*(\vec{N}); \vec{N})].$$

Thus, $\mathbb{E}[\text{cost}(\mathcal{A}(\vec{N}); \vec{N})] \geq (c\sqrt{m}/2)\text{cost}(S^*(\vec{N}); \vec{N})$ holds for at least one vector \vec{N} in the support of \mathcal{P} . So the algorithm \mathcal{A} must have an $\Omega(\sqrt{m})$ -approximation factor. This finishes the proof of Theorem 4.