On Facility Location Problem in the Local Differential Privacy Model

Vincent Cohen-Addad
Google Research

Yunus Esencayi
University at Buffalo

Chenglin Fan
UT Dallas

Marco Gaboradi
Boston University

Shi Li
University at Buffalo

Di Wang
KAUST

Abstract

We study the facility location problem under the constraints imposed by local differential privacy (LDP). Recently, Gupta et al. (2010) and Esencayi et al. (2019) proposed lower and upper bounds for the problem on the central differential privacy (DP) model where a trusted curator first collects all data and processes it. In this paper, we focus on the LDP model, where we protect a client's participation in the facility location instance. Under the HST metric, we show that there is a non-interactive ϵ-LDP algorithm achieving $O(n^{1/4}/\epsilon^2)$-approximation ratio, where n is the size of the metric. On the negative side, we show a lower bound of $\Omega(n^{1/4}/\sqrt{\epsilon})$ on the approximation ratio for any non-interactive ϵ-LDP algorithm. Thus, our results are tight up to a polynomial factor of ϵ. Moreover, unlike previous results, our results generalize to non-uniform facility costs.

1 INTRODUCTION

The *facility location problem* is a classical problem in combinatorial optimization and operations research, aimed at identifying where and how many facilities to open in order to satisfy requests from clients. This problem has been intensively studied starting from the 1960's (Kuehn and Hamburger, 1963; Manne, 1964; Stollsteimer, 1963) and has found several applications in Machine Learning, Data Mining and Bioinformatics (Arya et al., 2004; Jain and Vazirani, 2001; Charikar and Guha, 1999).

Proceedings of the 25[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

Formally, the problem can be defined as following.

Definition 1 (Facility (FL) Location Problem)**.** The input to the Facility Location (FL) problem is a tuple $(V, d, \vec{f}, \vec{b})$, where $V, |V| = n$ is the set of potential clients, (V, d) is a metric, $\vec{f} = (f_v)_{v\in V} \in \mathbb{R}^V_{\geq 0}$ is the facility cost for each location $v \in V$, and $\vec{b} = (b_v)_{v\in V} \in \{0,1\}^V_{\geq 0}$ indicates if each $v \in V$ is indeed a client. The goal is to find a set of facility locations $S \subseteq V$ which minimizes the following, where $d(v, S) = \min_{s\in S} d(v, s)$:

$$\mathrm{cost}_d(S; \vec{b}) := \sum_{s\in S} f_s + \sum_{v\in V} b_v d(v, S). \qquad (1)$$

The first term of (1) is called the *facility cost* and the second term is called the *connection cost*. We identify the problem as *uniform* facility location if $f_v = f$ for all $v \in V$.

Recently, several works have studied versions of this problem under the constraints imposed by Differential Privacy (DP) (Dwork et al., 2006) in order to provide provable protection to the privacy of the individual clients. Gupta et al. (2010) first studied the *uniform* facility location problem and showed that any DP algorithm that outputs the exact set of open facilities must have a (multiplicative) approximation ratio of $\Omega(\sqrt{n})$. This shows that there is no hope to get any useful information for the problem under DP constraints if we want the exact set of open facilities. For this reason, Gupta et al. and later Esencayi et al. (2019) considered the *super-set output setting*. Under this setting, instead of the exact set of open facilities, the output could be a super-set R of the set of open facilities — every client connects to the closest facility in R, and a facility is open if there is at least one client connected to it. Esencayi et al. showed that under the super-set output setting and the *hierarchically well-separated* *tree (HST) metrics*, 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. On the negative side, Esencayi et al. showed that, under the super-set output setting, 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.

Our contribution All previous works on differentially private facility location focused on the central model of differential privacy, where individuals' data are first collected and then processed. An alternative to the central model that has drawn much attention recently is the *local differential privacy (LDP) model*, where each individual manages his/her proper data and discloses them to a server through some differentially private mechanisms. The server collects the (now private) data of each individual and combines them into a resulting data analysis. A classical application of this model is the one aiming at collecting statistics from user devices like in the case of Google's Chrome browser (Erlingsson et al., 2014) and Apple's IOS (Tang et al., 2017).

Thus, a natural question is:

> *Problem 1:* Can we design accurate algorithms for facility location in the local model of differential privacy? What are the theoretical limitations for designing these algorithms in this model?

Moreover, all the previous works focused on the uniform cost setting, and they cannot be directly applied to the non-uniform setting. In particular, it is unclear whether the non-uniformity of the problem requires an additional price to pay in terms of privacy or accuracy. Thus another natural question is:

> *Problem 2:* Can we design a differentially private algorithm for the non-uniform setting with guarantees similar to the ones provided in the uniform setting?

Thus, in this work we focus on the facility location problem in the local differential privacy model with non-uniform costs. To our knowledge, we present the first result on the facility location problem in the LDP model. In our setting, every user in the metric has a private bit, which indicates if he/she participated in the facility location instance or not. We present an ϵ-LDP non-interactive algorithm which achieves an $O(\frac{n^{\frac{1}{4}}}{\epsilon^2})$ approximation ratio under the HST metric, where n is the size of the metric. To complement the result, we show a lower bound of $\Omega(\frac{n^{\frac{1}{4}}}{\sqrt{\epsilon}})$ on the approximation ratio of any non-interactive ϵ-LDP algorithm.

Finally, we remark that there is a flaw in Esencayi et al. (2019), in their analysis of the ϵ-DP $O\left(\frac{1}{\epsilon}\right)$-approximation algorithm for the uniform cost facility location problem in HST metrics. Therefore, both this result and the $O\left(\frac{\log n}{\epsilon}\right)$-approximation result for general metric are incorrect.

In this paper, we give an approximation ratio of $O\left(\frac{1}{\sqrt{\epsilon}}\right)$ for HST metrics, for the *non-uniform cost* facility location under the central ϵ-DP model and the superset output setting. Therefore, not only we fixed the issue in Esencayi et al. (2019), but also our algorithm works for the non-uniform facility cost case and matches the lower bound in Esencayi et al. (2019). Therefore, we apply the tree embedding result and obtain an approximation ratio of $O(\log n)$ for general metric. The details of the results are given in the supplementary material.

1.1 Related Work

Gupta et al. (2010) is the first paper studying the differentially private facility location problem. Under the ϵ-DP model, Gupta et al. 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 FL under general metrics that outputs the set of open facilities must have a (multiplicative) approximation ratio of $\Omega(\sqrt{n})$, which negatively shows that FL in DP model is useless. This motivates them to consider the superset output setting. In the same paper the authors showed that, under the setting, an $O(\frac{\log^2 n \log^2 \Delta}{\epsilon})$ approximation ratio is possible, where $\Delta = \max_{u,v \in V} d(u, v)$ is the diameter of the input metric.

Nissim et al. (2012) studied an abstract mechanism design model where DP is used to design approximately optimal mechanism, and they used facility location as one of their key examples. Besides the problem itself, the facility location problem has close connection to k-median clustering and submodular optimization, whose DP versions have been studied extensively. (Jones et al., 2020; Mitrovic et al., 2017; Cardoso and Cummings, 2019; Feldman et al., 2009; Gupta et al., 2010; Balcan et al., 2017; Perez-Salazar and Cummings, 2020).

The super-set output setting is the same as the problem in the Joint Differential Privacy model, which was introduced in Kearns et al. (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. JDP has been studied for many other combinatorial optimization problems (Hsu et al., 2016a,b; Huang and Zhu, 2018, 2019; Gupta et al., 2010; Jones et al., 2020).

Due to the space limit, some omitted proofs are included in the Supplementary Material.

2 PRELIMINARIES

2.1 Differentially Private Facility Location

Given a data range B and a dataset $\vec{b} = \{b_1, \cdots, b_n\} \in B^n$ where each record b_i belongs to a party i. Let $\mathcal{A} : B^n \mapsto \mathcal{S}$ be an algorithm on $\vec{b}$ that produces an output in $\mathcal{S}$. Let $\vec{b}_{-i}$ denote the vector $\vec{b}$ without entry of the party i. Also denote by $(v'_i, \vec{b}_{-i})$ the dataset by adding v'_i to $\vec{b}_{-i}$.

Definition 2 (Differential Privacy (Dwork et al., 2006))**.** A randomized algorithm $\mathcal{A}$ is ϵ-differentially private (DP) if for any $i \in [n]$, any two possible data entries $b_i, b'_i \in B$, any vector $\vec{b}_{-i} \in B^{[n]\setminus\{i\}}$ and for all events $\mathcal{T}$ in the output space of $\mathcal{A}$, we have $\Pr[\mathcal{A}(b_i, \vec{b}_{-i}) \in \mathcal{T}] \le e^\epsilon \Pr[\mathcal{A}(b'_i, \vec{b}_{-i}) \in \mathcal{T}]$.

For the facility location problem in the central model, we use $\vec{b} = (b_v)_{v\in V} \in \{0,1\}^V$ as input, where b_v indicates if v wants to be connected or not. Then ϵ-DP requires that for any input vectors $\vec{b}$ and $\vec{b}'$ with $|\vec{b} - \vec{b}'|_1 = 1$ and any event $\mathcal{T} \subseteq \mathcal{S}$, we have $\Pr[\mathcal{A}(\vec{b}) \in \mathcal{T}] \le e^\epsilon \Pr[\mathcal{A}(\vec{b}') \in \mathcal{T}]$.

In the super-set output setting for the problem, the output of an algorithm is a set $R \subseteq V$ of potential open facilities. Then, every client, or equivalently, every $v \in V$ with $b_v = 1$, will be connected to some facility in R using some rule (see Definition 7). Then the actual set S of open facilities is the set of locations in R with at least 1 connected client. Notice that the facility cost of S might be much smaller than that of R. This is why the super-set output setting may help in getting good approximation ratios.

Definition 3 (Local Differential Privacy (Dwork et al., 2006))**.** Consider n clients with each holding a private entry $b_i \in B$, and a server coordinating the protocol. An LDP protocol executes for some number T of rounds. In each round, the server sends a message, which is also called a query, to a subset of the clients requesting them to run a particular algorithm. Based on the query, each client i in the subset selects an algorithm, runs it on b_i, and sends the output back to the server.
A randomized algorithm $\mathcal{A}$ is ϵ-local differentially private (LDP) if for any client $i \in [n]$, any two possible data entries $b_i, b'_i \in B$ and for all events $\mathcal{T}$ in the output space of $\mathcal{A}$, we have $\Pr[\mathcal{A}(b_i) \in \mathcal{T}] \le e^\epsilon \Pr[\mathcal{A}(b'_i) \in \mathcal{T}]$. Moreover, if $T = 1$, we say that the protocol is **non-interactive**.

2.2 Hierarchical Well-Separated Tree Metrics and Related Notations

The classic result of Fakcharoenphol et al. (2004) 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 Gupta et al. (2010) and Esencayi et al. (2019), we reduce an arbitrary metric to a HST metric, with a loss of $O(\log n)$ in the approximation factor.

Definition 4. For any real number $\lambda > 1$, an integer $L \ge 1$, a λ-Hierarchically Well-Separated tree (λ-HST) of depth L is an edge-weighted rooted tree T satisfying the following properties:

1. Every root-to-leaf path in T has exactly L edges.
2. 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. Thus, the root r of T has level $\ell_T(r) = L$ and every leaf $v \in T$ has level $\ell_T(v) = 0$. [1] 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.

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. We guarantee that if a metric is a λ-HST metric, the correspondent λ-HST T is given.

We introduce some more useful definitions and tools. Let $(V, d, \vec{f}, \vec{b})$ be facility location instance such that (V, d) is a λ-HST metric. Let T be the correspondent λ-HST tree; so $V \subseteq V_T$ is the set of its leaves. Since we are dealing with this 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. We define a vector $\vec{N}$ over V_T: For every $u \in V_T$, let $N_u = \sum_{v\in T_u\cap V} b_v$ be the number of clients in the tree T_u.

[1] By scaling we assume the minimum non-zero distance in the metric is 1.

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 the cheapest 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. So, we can define f_u for any internal u to be the minimum of f_v over all descendants v of u.

Claim 5. With a loss of $O(1)$-factor in the approximation ratio, we can assume every $v \in V_T$ is a facility. Moreover, for every $u, v \in V_T$ such that v is a descendant of u, we have $f_u \leq f_v$.

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

Definition 6. For a set $M \subseteq V_T$ of vertices in T, let

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

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, where the expectation is over the randomness of $\mathcal{A}$.

3 BASE ALGORITHM FOR FACILITY LOCATION ON HST METRICS

In this section, we give a base algorithm without any privacy guarantee as the starting point for the ϵ-LDP algorithm in the local model. The main idea behind the algorithm is similar to that of Esencayi et al. (2019). For every vertex v, we compare the cost of opening v and that of connecting all clients in T_v to v. If the former is smaller, then we mark v. We can show that the min-set of all marked facilities gives an $O(1)$-approximation to the facility location problem. However, to allow an easy transition from the base algorithm to the one with DP guarantee, we make it more general and involved. The parameter $\lambda > 1$ is a constant. Its precise value is not important for our LDP algorithm in Section 4. For the $O(\frac{1}{\sqrt{\epsilon}})$-approximate DP algorithm in Appendix B, the only requirement is that $\lambda < 2$.

3.1 Description of Algorithm and Useful Definitions

Before describing the algorithm, we need to make a rule on how we connect clients to open facilities. Instead of connecting each client to its closest open facility using the tree metric, it is more convenient for us to connect it to the *genetically closest facility*:

Definition 7. Given a non-empty set $R \subseteq V_T$ of facilities, and a client $v \in V$, we define the *genetically closest facility of v in R* to be the facility $u \in R$ with the lowest common ancestor (LCA) of u and v being the lowest, breaking ties using a predefined total order over facilities.

The genetically closest facility of v may not be the same as its closest facility according to the metric if $\lambda < 2$. However, they are equivalent up to a factor of 2. Using genetically closest facilities turns out to be more convenient for us.

Suppose we are given any non-empty set $R \subseteq V_T$ of facilities and we connect each client to its genetically closest facility in R. In our super-set output setting, we only open the facilities that have connected clients. We use open(R) to denote this set.

Algorithm 1 FL-tree(ρ, ρ', τ) $\quad \rho, \rho' \geq 1, \tau \in \{\frac{1}{2}, 1, 2\}$

▷ This is also called the base algorithm.

1: Let $M \leftarrow \left\{v \in V_T : \lambda^{\ell(v)} \geq \frac{f_v}{\rho} \text{ or } N_v \cdot \lambda^{\ell(v)} \geq \tau\rho' f_v\right\}$. ▷ We call facilities in M *marked* and other facilities *unmarked*
2: $R \leftarrow$ min-set(M)
3: **return** R but only open $S :=$ open(R)

The base algorithm FL-tree (Algorithm 1) takes three parameters $\rho, \rho' \geq 1$ and $\tau \in \{\frac{1}{2}, 1, 2\}$. Recall that $N_u = \sum_{v \in T_u \cap V} b_v$ is the number of clients in the tree T_u. To get an $O(1)$-approximation, we can simply set $\rho = \rho' = \tau = 1$. The parameters ρ and ρ' are introduced for easy comparisons with the DP algorithms. In the analysis, we may compare a DP algorithm with $\tau = 1$ with the base algorithm with $\tau = 1/2$ or $\tau = 2$; this is the reason we introduce the parameter.

Definition 8. Let $\rho \geq 1$ be fixed. We say a vertex v is *cheap* if $\lambda^{\ell(v)} \geq \frac{f_v}{\rho}$. Otherwise, we say v is *expensive*.

We may assume the root of T is cheap, by extending the tree at the root: This operation adds a new root and lets the old root be a child of the new one. Whether a vertex is cheap or expensive only depends on the metric and the facility costs. It does not depend on the client set, i.e, the N_u or b_v values. The definition depends on ρ, but we guarantee that ρ will be clear from the context.

In Algorithm 1, we *mark* some vertices in V_T and let M be those vertices. By definition 8, all cheap vertices will be marked. The vertices that are not marked are said to be *unmarked*. The algorithm returns the set $R := \text{min-set}(M)$ and opens the set $S := \text{open}(R)$.

Notice that cheap and marked vertices satisfy the following monotonicity property: An ancestor of any cheap (marked) vertex is also cheap (marked). This holds since along a root-to-leaf path, the facility costs are non-decreasing and N_v values are non-increasing. Due to the properties, we say

- a cheap vertex v is *minimal-cheap* if all of its children are expensive,
- an expensive vertex v is *maximal-expensive* if its parent is cheap,
- a marked vertex $v \in V_T$ *minimal-marked* if all its children are unmarked, and
- an unmarked vertex $v \in V_T$ *maximal-unmarked* if its parent is marked.

By the definition of min-set, the R returned by Algorithm 1 is exactly the set of minimal-marked vertices.

Definition 9. For any $v \in V_T$, let $B_v = \min\{N_v \cdot \lambda^{\ell(v)}, f_v\}$.

The B_v's will be used as budgets to pay the costs:

Claim 10. In any solution to the FL instance, the cost of open facilities in T_v plus the connection cost of clients in T_v is at least B_v.

Proof. This holds since either some facility in T_v is open, which costs at least f_v, or each client in T_v have connection cost at least $\lambda^{\ell(v)}$. □

Let opt be the cost of the optimum solution. Thus the following corollary is immediate:

Corollary 11. For any $V_1 \subseteq V_T$ which does not contain an ancestor-descendant pair, we have opt $\geq \sum_{v \in V_1} B_v$.

Analysis of facility cost The analysis of facility cost is straightforward.

Claim 12. For any maximal-unmarked vertex $v \in V_T$, the total connection cost for the clients in T_v in the base algorithm is no more than $O(\rho') \cdot B_v$.

Proof. Recall that both λ and τ are constants. Since v is maximal-unmarked, its parent u is marked. Then all the clients in T_v will be connected to some facility in T_u. So, their total connection cost is at most $N_v \cdot O(1) \cdot \lambda^{\ell(v)+1} \leq O(\rho') f_v$. As $B_v = \min\{f_v, N_v \cdot \lambda^{\ell(v)}\}$, the cost is at most $O(\rho') \cdot B_v$. □

Corollary 13. The connection cost of the solution produced by Algorithm 1 is at most $O(\rho') \cdot \text{opt}$.

Proof. Let MU be the set of maximal-unmarked vertices in V_T. Note that again there is no ancestor-descendant pairs in MU and every $v \in V$ has exactly one ancestor in the set MU. By Lemma 12, the connection cost is at most

$$O(\rho') \sum_{v \in \text{MU}} B_v \leq O(\rho')\text{opt}. \qquad \square$$

3.2 Analysis of facility cost

In this section, we analyze the facility cost of the base algorithm, which is much more involved.

Definition 14. Let u^* be the maximal-expensive vertex such that $N_{u^*} \geq 1$ with the largest $\ell(u^*)$. Let $\ell^* = \ell(u^*)$.

Lemma 15. Let u be any maximal-expensive vertex with $\ell(u) > \ell^*$. Then any algorithm using the genetically closest vertex rule will open no facilities in T_u.

Proof. Notice that $N_u = 0$ by the definition u^* and ℓ^* and that $\ell(u) > \ell^*$. Assuming some client $v \in V$ is connected to some facility inside T_u, and v is in $T_{u'}$ for some maximal-expensive vertex u'. Then $\ell(u') \leq \ell^* < \ell(u)$. Since u and u' do not have ancestor-descendant relation, the LCA of u and u' has level at least $\ell(u) + 1 \geq \ell(u') + 2$. However, the parent u'' of u' is cheap and has level $\ell(u') + 1$. So, v must be connected to a facility inside $T_{u''}$. □

Therefore, for any maximal expensive vertex u with $\ell(u) > \ell^*$, we have $N_u = 0$ and no facilities in T_u will be open. Thus, for the purpose of analysis, we can remove T_u from T. (This may decrease n; but it can only make the performance better.) So we have the following claim:

Claim 16. Every expensive vertex u has $\ell(u) \leq \ell^*$.

Corollary 17. opt $\geq \lambda^{\ell^*}$.

Proof. We know that T_{u^*} contains at least one client. So we have opt $\geq B_{u^*} = \min\{f_{u^*}, N_{u^*}\lambda^{\ell^*}\} \geq \lambda^{\ell^*}$ since $N_{u^*} \geq 1$ and $f_{u^*} > \rho \cdot \lambda^{\ell^*} \geq \lambda^{\ell^*}$ as u^* is expensive. □

The following lemma and corollary are more general than needed in this section, but they will be useful in analyzing algorithms in Section 4 and Appendix B.

Lemma 18. Let M be the set of marked vertices as in the base algorithm, $M' \subseteq V_T$ be any subset containing all cheap vertices. Let $R' = \text{min-set}(M')$ and $S' = \text{open}(R')$. Then, we have

$$\sum_{u \in S' \cap M} f_u \leq O(\rho) \cdot \text{opt}.$$

Proof. We will consider the expensive and cheap vertices in $S' \cap M$ separately to bound the total facility cost.

First assume that u is an expensive vertex. Notice that $u \in S' \cap M \Rightarrow u \in M \Rightarrow N_u \cdot \lambda^{\ell(u)} \geq \tau\rho' f_u \geq \tau f_u$. This is true since $\rho' \geq 1$. Then, we clearly have $f_u \leq O(1) \cdot B_u$. S' does not have ancestor-descendant pairs, so the cost of expensive vertices in $S' \cap M$ can be bounded by $O(1) \cdot \mathrm{opt}$ by Corollary 11.

Assume that u is a cheap vertex and $N_u > 0$. Then, $\lambda^{\ell(u)} \geq \frac{f_u}{\rho} \Rightarrow f_u \leq \rho \cdot \lambda^{\ell(u)} \leq \rho \cdot N_u \cdot \lambda^{\ell(u)}$. Since $\rho \geq 1$, we also know $f_u \leq \rho \cdot f_u$. Then, $f_u \leq \rho \cdot B_u$. So, the cost of cheap vertices in $u \in S' \cap M$ with $N_u > 0$ can be bounded by $O(\rho) \cdot \mathrm{opt}$ by Corollary 11.

Finally, let us analyze the cost of cheap vertices $u \in S' \cap M$ with $N_u = 0$. Focus on such a u. Since $u \in S'$, there must be some v which connects to u through its parent edge. Let the least common ancestor of u and v be u'', and let u' be a child of u'' such that v is in $T_{u'}$. Note that u and u'' are cheap, u' is maximal-expensive, and $\ell(u) \leq \ell(u')$. Then $B_{u'} = \min\{f_{u'}, N_{u'}\lambda^{\ell(u')}\} \geq \lambda^{\ell(u')} \geq \lambda^{\ell(u)} \geq \frac{f_u}{\rho}$. So $f_u \leq \rho \cdot B_{u'}$. We charge f_u using $B_{u'}$. As we are using a consistent way to break ties when connecting clients, we will not use $B_{u'}$ for the same u' to charge f_u for many different u's. Also, all the u''s are maximal-expensive and they do not contain ancestor-descendant pairs. So, the cost of cheap vertices $u \in S' \cap M$ with $N_u = 0$ can be bounded by $O(\rho) \cdot \mathrm{opt}$. □

Corollary 19. The set S of open facilities produced by Algorithm 1 has facility cost at most $O(\rho) \cdot \mathrm{opt}$.

Proof. We apply Claim 18 with $M' = M$, $R' = R$ and $S' = S \subseteq M$. The facility cost for S is at most $O(\rho) \cdot \mathrm{opt}$. □

4 ϵ-LDP $O(\frac{n^{1/4}}{\epsilon^2})$-APPROXIMATION ALGORITHM

In this section, we consider the facility location problem in the LDP model and propose the first algorithm. In this local model we assume every vertex $v \in V$ is a potential client and the location of v is public. However, each v has a private bit $b_v \in \{0, 1\}$ indicating if she/he wants to be connected or not. In other words, b_v indicates if v is indeed a client or not in the facility location instance. Let $n = |V|$. We first provide an algorithm and show the upper bound of the utility of its output. Our algorithm is based on the random response mechanism and our previous framework in Section 3.

Theorem 20. Under the super-set output setting, there exists an ϵ-LDP algorithm (Algorithm 2) for n potential clients in the HST metrics with the (expected) cost of $O(\frac{n^{\frac{1}{4}}}{\epsilon^2}) \cdot \mathrm{opt}$, where opt is the optimal cost.

Throughout this section, we fix $\rho = \rho' = n^{\frac{1}{4}}$. The value of λ does not matter much and so we fix it to 2; but we keep λ for notation consistency. For the privacy part in our algorithm, each user just perturbs his/her bit b_v and get a private one b'_v by using the random response mechanism to ensure ϵ-LDP. For the utility part, we start from the base algorithm FL-tree($\rho = \rho' = n^{\frac{1}{4}}, \tau = 1$) in Algorithm 1, and then add noises to the N_v variables. We use $\tilde{N}_v$ to denote the noisy version of N_v, and let M', R' and S' correspond to M, R and S, to avoid confusion. Then, vertices in M' are called *noisily-marked*, and the others are called *noisily-unmarked*.

Algorithm 2 LDP-FL-tree(ϵ)

On the user's side:

for every $v \in V$ **do**

  Get a $b'_v \in \{0, 1\}$ where $b'_v \leftarrow b_v$ with probability $\frac{e^\epsilon}{e^\epsilon+1}$; $b'_v \leftarrow 1 - b_v$ with probability $\frac{1}{e^\epsilon+1}$.

end for

Each user $v \in V$ sends b'_v to the server.

On the server's side:

for every $v \in V_T$ do:

  $\tilde{N}_v \leftarrow \frac{e^\epsilon+1}{e^\epsilon-1}|b'(V \cap T_v)| - \frac{1}{e^\epsilon+1}|V \cap T_v|$

let $M' \leftarrow \left\{v \in V_T : \lambda^{\ell(v)} \geq \frac{f_v}{n^{1/4}} \text{ or } \tilde{N}_v \cdot \lambda^{\ell(v)} \geq n^{\frac{1}{4}} \cdot f_v\right\}$

  ▷ Verticies in M' are said to be *noisily-marked*, and the other verticies are *noisily-unmarked*.

$R' \leftarrow \text{min-set}(M')$

return R' but only open $S' := \text{open}(R')$

In the algorithm, $b'(V \cap T_v) = \sum_{u \in V \cap T_v} b'_u$ is the total b' value over all leaves of T_v.

4.1 Analysis of Utility

We proceed to consider the utility of the algorithm. As we are using $\tilde{N}_v$ to replace N_v in the base algorithm, we need to make sure $\mathbb{E}[\tilde{N}_v] = N_v$ for every $v \in T$. The following claim shows that $\tilde{N}_v$ is an unbiased estimator for N_v and its variance is not large. It's proof directly follows from the random response mechanism.

Claim 21. For every $v \in V$, we have $\mathbb{E}[\tilde{N}_v] = N_v$ and $\mathrm{Var}[\tilde{N}_v] = \frac{e^\epsilon}{(e^\epsilon-1)^2}|T_v \cap V|$.

To obtain the approximation ratio of Algorithm 2, we will compare it with the base algorithm FL-tree (Algorithm 1) with $\rho = \rho' = n^{1/4}$ and $\tau = \frac{1}{2}$ or $\tau = 2$, whose solution has a cost $O(n^{1/4}) \cdot \mathrm{opt}$. We will analyze the facility and connection costs of Algorithm 2 separately.

Facility Cost of Algorithm 2 We compare the extra facility cost incurred by Algorithm 2, to that of the base algorithm with $\rho = \rho' = n^{1/4}$ and $\tau = \frac{1}{2}$.

We break S' into two parts: $S' \cap M$ and $S' \setminus M$. By Lemma 18, the cost of $S' \cap M$ is at most $O(\rho) \cdot \text{opt}$. Therefore, it suffices for us to bound the cost of $S' \setminus M \subseteq M' \setminus M$, *i.e.,* the unmarked but noisily-marked facilities.

Now we consider a vertex $v \notin M$; that is, the vertex v satisfies $\lambda^{\ell(v)} < \frac{f_v}{n^{\frac{1}{4}}}$ and $N_v \lambda^{\ell(v)} < \frac{n^{\frac{1}{4}} f_v}{2}$. Using Chebyshev's inequality, we can get a bound of the probability that $v \in M'$, *i.e.,* $\tilde{N}_v \lambda^{\ell(v)} \geq n^{\frac{1}{4}} f_v$:

$$\Pr[v \in M'] = \Pr\left[\tilde{N}_v \geq \frac{n^{\frac{1}{4}} f_v}{\lambda^{\ell(v)}}\right]$$

$$\leq \Pr\left[\tilde{N}_v \geq N_v + \frac{n^{\frac{1}{4}} f_v}{2 \cdot \lambda^{\ell(v)}}\right] \leq \frac{\text{Var}[\tilde{N}_v]}{\left(\frac{n^{\frac{1}{4}} f_v}{2 \cdot \lambda^{\ell(v)}}\right)^2} = \frac{\frac{e^\epsilon |T_v \cap V|}{(e^\epsilon - 1)^2}}{\left(\frac{n^{\frac{1}{4}} f_v}{2 \cdot \lambda^{\ell(v)}}\right)^2}$$

$$= \frac{4e^\epsilon |T_v \cap V| \lambda^{2\ell(v)}}{(e^\epsilon - 1)^2 \sqrt{n} f_v^2} \leq \frac{4e^\epsilon |T_v \cap V| \lambda^{\ell(v)}}{(e^\epsilon - 1)^2 N^{3/4} f_v},$$

where the second equality is due to Claim 21 and the last inequality is due to the fact used that $\lambda^{\ell(v)} < \frac{f_v}{n^{1/4}}$. Thus, in total we have

$$\Pr[v \in M'] \cdot f_v \leq \frac{4e^\epsilon |T_v \cap V| \lambda^{\ell(v)}}{(e^\epsilon - 1)^2 N^{3/4}}. \tag{2}$$

Since we have that $\lambda^{\ell(v)}$ goes down exponentially along a root-to-leaf path, and $|T_v \cap V|$ is the number of leaves in the tree T_v. Therefore, a simple argument could show that the sum of the right side of (2) over all $v \notin M$, is at most $\frac{\lambda}{\lambda - 1}$ times the sum over all maximal-unmarked vertices. Notice that an unmarked vertex is expensive and by Claim 16 all expensive vertices have level at most ℓ^*. Therefore, the sum of (2) over all $v \notin M$ is at most $\frac{\lambda}{\lambda-1} \cdot \frac{4e^\epsilon n \lambda^{\ell^*}}{(e^\epsilon - 1)^2 N^{\frac{3}{4}}} = O(\frac{n^{\frac{1}{4}}}{\epsilon^2}) \lambda^{\ell^*}$.

Finally, we notice that from Corollary 17, we can get $\text{opt} \geq \lambda^{\ell^*}$. Therefore, the expectation of facility cost of $S' \setminus M \subseteq M' \setminus M$ is at most $O(\frac{n^{\frac{1}{4}}}{\epsilon^2})\text{opt}$.

Connection Cost of Algorithm 2 For the analysis of the extra connection cost, we will compare our Algorithm 2 with the base algorithm with $\rho = \rho' = n^{\frac{1}{4}}$ and $\tau = 2$. Again, M, R and S are as in the base algorithm, and M', R' and S' are as in the LDP algorithm.

It is convenient to assume that in each of the two algorithms, every client is connected to its lowest marked (noisily-marked) ancestor. Notice that the actual connection costs may be larger by a factor of 2, which can be ignored. We know that connecting all clients to their respective lowest marked ancestors has a cost of $O(n^{\frac{1}{4}}) \cdot \text{opt}$. Now, in Algorithm 2, every client is connected to its respective lowest noisily-marked ancestors.

To bound the extra connection cost, for every $v \in M \setminus M'$, we impose a cost of moving the connection of all clients in T_v from v to its parent. Notice that we can assume v is expensive since otherwise $v \in M'$.

For a fixed expensive $v \in M$, we first bound the probability of the event $v \notin M'$. Notice that we have $\lambda^{\ell(v)} < \frac{f_v}{n^{1/4}}$ and $N_v \cdot \lambda^{\ell(v)} \geq 2N^{1/4} f_v$, which imply $N_v \geq \frac{2N^{\frac{1}{4}} f_v}{\lambda^{\ell(v)}} \geq \frac{2N^{\frac{1}{4}} f_v}{f_v / n^{1/4}} = 2\sqrt{n}$. Thus we have

$$\Pr[v \notin M'] = \Pr\left[\tilde{N}_v < \frac{n^{\frac{1}{4}} f_v}{\lambda^{\ell(v)}}\right] \leq \Pr\left[\tilde{N}_v \leq N_v - \frac{N_v}{2}\right]$$

$$\leq \frac{\text{Var}[\tilde{N}_v]}{\left(\frac{N_v}{2}\right)^2} = \frac{4e^\epsilon}{(e^\epsilon - 1)^2} \cdot \frac{|T_v \cap V|}{N_v^2}.$$

Thus, in expectation, the cost of the reconnecting operation for v will be at most

$$\Pr\left[v \notin M'\right] \cdot N_v \cdot O(1) \cdot \lambda^{\ell(v)} \leq O\left(\frac{1}{\epsilon^2}\right) \cdot \frac{|T_v \cap V|}{N_v} \cdot \lambda^{\ell(v)}$$

$$\leq O\left(\frac{1}{\epsilon^2}\right) \cdot \frac{|T_v \cap V|}{\sqrt{n}} \cdot \lambda^{\ell(v)}. \tag{3}$$

Notice that $T_v \cap V$ is the number of leaves in T_v and $\lambda^{\ell(v)}$ decreases exponentially as the level goes down. So, it is easy to see that the sum of (3) over all expensive marked vertices v is upper bounded by $\frac{\lambda}{\lambda-1}$ times the sum over all maximal-expensive marked vertices.

For a maximal-expensive marked vertex v, we have

$$O\left(\frac{1}{\epsilon^2}\right) \cdot \frac{|T_v \cap V|}{\sqrt{n}} \cdot \lambda^{\ell(v)} \leq O\left(\frac{1}{\epsilon^2}\right) \cdot \frac{|T_v \cap V|}{\sqrt{n}} \cdot \frac{f_v}{n^{\frac{1}{4}}}$$

$$= O\left(\frac{1}{\epsilon^2}\right) \cdot |T_v \cap V| \cdot \frac{f_v}{n^{\frac{3}{4}}}.$$

Let v^* be the maximal-expensive marked vertex with the maximum cost f_{v^*}. Then, the sum of (3) over all expensive marked vertices v is at most $O\left(\frac{1}{\epsilon^2}\right) \cdot n \cdot \frac{f_{v^*}}{n^{\frac{3}{4}}} = O\left(\frac{n^{\frac{1}{4}}}{\epsilon^2}\right) \cdot f_{v^*}$. Notice that $\text{opt} \geq B_{v^*} = \min\{f_{v^*}, N_{v^*} \lambda^{\ell(v^*)}\} = f_{v^*}$, where the equality holds when v^* is expensive but marked. Thus the expectation of extra connection cost of Algorithm 2 compared to the base algorithm is $O\left(\frac{n^{\frac{1}{4}}}{\epsilon^2}\right) \cdot \text{opt}$.

5 Lower Bound of Non-interactive ϵ-LDP Algorithms

In this section, we give a lower bound of $\Omega(n^{\frac{1}{4}}\epsilon^{-\frac{1}{2}})$ on the utility of any non-interactive ϵ-LDP algorithm. To do so, we focus on the following standalone problem in the bulk of the section. Suppose they are n parties indexed by $[n]$, each player $i \in [n]$ having an input $X_i \in \{0,1\}$. Let $\epsilon \in (0,1)$, these n parties need to run an ϵ-LDP algorithm. We are promised that we are in one of the following two cases, where $c > 0$ is a small enough absolute constant:

- Case (a): $X_i = 0$ for every party i.
- Case (b): $X_i \sim \text{Bern}\left(\frac{c}{(e^\epsilon-1)\sqrt{n}}\right)$ for every party i, where we use $\text{Bern}(p)$ to denote the Bernoulli distribution with mean p.

The goal of the problem for the central server is to decide which of the two cases we are at. We say an algorithm succeeds if the central server outputs correctly the case number. We first prove the following theorem:

Theorem 22. For a small enough constant $c > 0$, there is no non-interactive ϵ-LDP algorithm that can succeed with probability more than 0.6.

Before showing the proof, let us first see how Theorem 22 implies a lower bound of our problem. We consider a facility location instance with two points u and v in the metric, where the distance between u and v is $n^{\frac{1}{4}}$. All the n clients are collocated at u. Moreover, v has a facility of cost 0 and u has a facility of cost $n^{\frac{1}{2}}\epsilon^{-\frac{1}{2}}$. Suppose we are in case (a) or case (b) as defined above. A client i participates in the facility location instance if and only if $x_i = 1$. To disallow the algorithm to take the advantage of the super-set setting, we place another client at u which is always present.

Thus, in case (a), the optimum solution does not open u and its cost is $n^{\frac{1}{4}}$, since if our algorithm opens u, the cost will be $\frac{n^{\frac{1}{2}}}{\sqrt{\epsilon}}$. In case (b), an optimum solution can open u and its cost is $\frac{n^{\frac{1}{2}}}{\sqrt{\epsilon}}$. Since if our algorithm does not open u, the (expected) cost will be $\Theta(\frac{\sqrt{n}}{\epsilon}\cdot n^{\frac{1}{4}}) = \Theta(\frac{n^{\frac{3}{4}}}{\epsilon})$. By Theorem 22, our algorithm will make a mistake with constant probability in at least one of the two cases. And for each case, its approximation ratio to the optimal cost is $\frac{n^{\frac{1}{4}}}{\sqrt{\epsilon}}$. Thus, for any ϵ non-interactive LDP algorithm, its approximation ratio should be at least $\Omega(\frac{n^{\frac{1}{4}}}{\sqrt{\epsilon}})$.

5.1 Proof of Theorem 22

Before the prove, we first prove the theorem for a very specific algorithm, where each player i sends a noisy bit X'_i of X_i to the central server. That is, we let $X'_i = X_i$ with probability $\frac{e^\epsilon}{e^\epsilon+1}$ and $X'_i = 1-X_i$ with probability $\frac{1}{e^\epsilon+1}$. Then the central server has to output the case number based on $(X'_1, X'_2, \cdots, X'_n)$. We call such an algorithm a *canonical algorithm*.

We set up some notations first. We define $X = (X_1, X_2, \cdots, X_n)$ and $X' = (X'_1, X'_2, \cdots, X'_n)$. Then, in case (a), we have $X'_i \sim \text{Bern}(a)$, where $a := \frac{1}{e^\epsilon+1}$. In case (b), we have $X'_i \sim \text{Bern}(b)$, where $b := \left(1 - \frac{c}{(e^\epsilon-1)\sqrt{n}}\right)\cdot\frac{1}{e^\epsilon+1} + \frac{c}{(e^\epsilon-1)\sqrt{n}}\cdot\frac{e^\epsilon}{e^\epsilon+1} = \frac{1}{e^\epsilon+1} + \frac{e^\epsilon-1}{e^\epsilon+1}\cdot\frac{c}{(e^\epsilon-1)\sqrt{n}} = \frac{1}{e^\epsilon+1} + \frac{c}{(e^\epsilon+1)\sqrt{n}} = a\left(1+\frac{c}{\sqrt{n}}\right)$. We use $\Pr_a, \Pr_b$ denotes the probabilities under cases (a) and (b) respectively. To prove that a canonical algorithm can not succeed with probability more than 0.6, we first prove that the statistical distance between the distribution for X' under case (a) and that under case (b) is small, whose proof is given in Appendix:

Lemma 23. $\sum_{x'\in\{0,1\}^n}\left|\Pr_a[X'=x'] - \Pr_b[X'=x']\right| \le 0.4.$

To show how Lemma 23 implies that the previous canonical algorithm can not succeed with probability more than 0.6, we assume the adversary chooses case (a) and case (b) with probability $\frac{1}{2}$ for each. To succeed with the largest probability, the algorithm should output 'a' if $\Pr_a[X'=x'] \ge \Pr_b[X'=x']$ if it sees $X' = x'$ and 'b' otherwise. Thus, the success probability of the algorithm is

$$\sum_{x'\in\{0,1\}^n}\frac{1}{2}\max\left\{\Pr_a[x'], \Pr_b[x']\right\}$$
$$= \frac{1}{2}\sum_{x'\in\{0,1\}^n}\left(\frac{\Pr_a[x'] + \Pr_b[x']}{2} + \frac{\left|\Pr_a[x'] - \Pr_b[x']\right|}{2}\right)$$
$$= \frac{1}{2} + \frac{1}{4}\sum_{x'\in\{0,1\}^n}\left|\Pr_a[x'] - \Pr_b[x']\right| \le 0.6.$$

Above, we used $\Pr_a[x']$ for $\Pr_a[X'=x']$ and $\Pr_b[x']$ for $\Pr_b[X'=x']$.

Now we proceed to consider any general non-interactive algorithm $\mathcal{A}$. Assume in $\mathcal{A}$ every player i sends a message Y_i to the central server. Then the algorithm has to output the case number based on the messages $(Y_1, Y_2, \cdots, Y_n)$. We need to show that if the algorithm is ϵ-locally differentially private, then the algorithm can not succeed with probability more than 0.6.

Indeed, we show that it suffices for each player i to send the bit X'_i generated as in a canonical algorithm to the central server for it to simulate the algorithm $\mathcal{A}$. To prove the statement, we fix a player i. Let $P_0(\cdot)$ and $P_1(\cdot)$ be the probability measurement functions for Y_i conditioned on that $X_i = 0$ and $X_i = 1$ respectively. That is, for every measurable set S, we have

$$\Pr[Y_i \in S | X_i = 0] = P_0(S),$$
$$\Pr[Y_i \in S | X_i = 1] = P_1(S).$$

Since the algorithm is ϵ-LDP, we must have $e^{-\epsilon} \leq \frac{P_0(S)}{P_1(S)} \leq e^{\epsilon}$.

For convenience, we define $p = \frac{c}{(e^\epsilon - 1)\sqrt{n}}$; so $X_i \sim \mathrm{Bern}(p)$ under case (b). Then $a = \frac{1}{e^\epsilon + 1}$ and $b = a + \frac{(e^\epsilon - 1)p}{e^\epsilon + 1}$. We have for every measurable set S,

$$\begin{aligned}
&\Pr_a[Y_i \in S] = P_0(S),\\
&\Pr_b[Y_i \in S] = (1-p)P_0(S) + p \cdot P_1(S)\\
&= P_0(S) + p \cdot (P_1(S) - P_0(S)).
\end{aligned}$$

We prove the following lemma to establish the reduction from general non-interactive algorithms to a canonical one:

Lemma 24. There are probability measurement functions P'_0 and P'_1 such that

$$\begin{aligned}
&P_0(S) = (1-a)P'_0(S) + aP'_1(S)\\
&= P'_0(S) + a(P'_1(S) - P'_0(S)),\\
\text{and}\quad &P_0(S) + p \cdot (P_1(S) - P_0(S))\\
&= (1-b)P'_0(S) + bP'_1(S)\\
&= P'_0(S) + b(P'_1(S) - P'_0(S)).
\end{aligned}$$

Now in the new algorithm $\mathcal{A}'$, player i will send X'_i generated as in a canonical algorithm to the central server. We use the two probability measurement functions $P'_0(\cdot)$ and $P'_0(\cdot)$ from the above lemma. If the central server sees $X'_i = 0$, it produces Y_i according to P'_0. If $X'_i = 1$, it produces Y_i according to P'_1. By the lemma, the distribution for the Y_i generated by the central server in $\mathcal{A}'$ will be the same as the distribution for Y_i in $\mathcal{A}$.

Therefore, we can simulate $\mathcal{A}$ using the algorithm $\mathcal{A}'$. However, $\mathcal{A}'$ is a canonical algorithm and thus can not succeed with probability more than 0.6. This finishes the proof of Theorem 22.

References

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

Balcan, M.-F., Dick, T., Liang, Y., Mou, W., and Zhang, H. (2017). 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.

Cardoso, A. R. and Cummings, R. (2019). Differentially private online submodular minimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1650–1658.

Charikar, M. and Guha, S. (1999). Improved combinatorial algorithms for the facility location and k-median problems. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pages 378–388. IEEE.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284. Springer.

Erlingsson, Ú., Pihur, V., and Korolova, A. (2014). Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067.

Esencayi, Y., Gaboardi, M., Li, S., and Wang, D. (2019). Facility location problem in differential privacy model revisited. *Advances in neural information processing systems.*

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

Feldman, D., Fiat, A., Kaplan, H., and Nissim, K. (2009). Private coresets. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 361–370. ACM.

Gupta, A., Ligett, K., McSherry, F., Roth, A., and Talwar, K. (2010). 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.

Hsu, J., Huang, Z., Roth, A., Roughgarden, T., and Wu, Z. S. (2016a). Private matchings and allocations. *SIAM Journal on Computing*, 45(6):1953–1984.

Hsu, J., Huang, Z., Roth, A., and Wu, Z. S. (2016b). 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.

Huang, Z. and Zhu, X. (2018). 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.

Huang, Z. and Zhu, X. (2019). Scalable and jointly differentially private packing. *arXiv preprint arXiv:1905.00767*.

Jain, K. and Vazirani, V. V. (2001). 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.

Jones, M., Nguyen, H. L., and Nguyen, T. (2020). Differentially private clustering via maximum coverage. *arXiv preprint arXiv:2008.12388*.

Kearns, M., Pai, M., Roth, A., and Ullman, J. (2014). Mechanism design in large games: Incentives and privacy. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pages 403–410. ACM.

Kuehn, A. A. and Hamburger, M. J. (1963). A heuristic program for locating warehouses. *Management science*, 9(4):643–666.

Manne, A. S. (1964). Plant location under economies-of-scale—decentralization and computation. *Management Science*, 11(2):213–235.

Mitrovic, M., Bun, M., Krause, A., and Karbasi, A. (2017). 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.

Nissim, K., Smorodinsky, R., and Tennenholtz, M. (2012). Approximately optimal mechanism design via differential privacy. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 203–213.

Perez-Salazar, S. and Cummings, R. (2020). Differentially private online submodular maximization. *arXiv preprint arXiv:2010.12816*.

Stollsteimer, J. F. (1963). A working model for plant numbers and locations. *Journal of Farm Economics*, 45(3):631–645.

Tang, J., Korolova, A., Bai, X., Wang, X., and Wang, X. (2017). Privacy loss in apple's implementation of differential privacy on macos 10.12. *arXiv preprint arXiv:1709.02753*.

Supplementary Material: On Facility Location Problem in the Local Differential Privacy Model

A MISSING PROOFS

In this section, we provide the missing proofs which were not given in the main paper due to the page limit.

Claim 21. For every $v \in V$, we have $\mathbb{E}[\tilde{N}_v] = N_v$ and $\mathrm{Var}[\tilde{N}_v] = \frac{e^\epsilon}{(e^\epsilon-1)^2}|T_v \cap V|$.

Proof. We denote $V_v = T_v \cap V$, for any fixed $v \in V$ we have

$$\begin{aligned}
\mathbb{E}\left[\tilde{N}_v\right] &= \frac{e^\epsilon+1}{e^\epsilon-1}\left(\mathbb{E}\left[|D' \cap V_v|\right] - \frac{1}{e^\epsilon+1}|V_v|\right) \\
&= \frac{e^\epsilon+1}{e^\epsilon-1}\left(\frac{e^\epsilon}{e^\epsilon+1}|D \cap V_v| + \frac{1}{e^\epsilon+1}|V_v \setminus D| - \frac{1}{e^\epsilon+1}|V_v|\right) \\
&= \frac{e^\epsilon+1}{e^\epsilon-1} \cdot \frac{e^\epsilon-1}{e^\epsilon+1}|D \cap V_v| = |D \cap V_v| = N_v. \\
\mathrm{Var}[\tilde{N}_v] &= \left(\frac{e^\epsilon+1}{e^\epsilon-1}\right)^2 \mathrm{Var}[|D' \cap V_v|] = \left(\frac{e^\epsilon+1}{e^\epsilon-1}\right)^2 \sum_{u \in V_v} \mathrm{Var}[1_{u \in D'}] \\
&= \left(\frac{e^\epsilon+1}{e^\epsilon-1}\right)^2 \cdot \frac{e^\epsilon}{e^\epsilon+1} \cdot \frac{1}{e^\epsilon+1} \cdot |V_v| = \frac{e^\epsilon}{(e^\epsilon-1)^2}|V_v|.
\end{aligned}$$

□

Lemma 23. $\sum_{x' \in \{0,1\}^n} \left|\Pr_a[X' = x'] - \Pr_b[X' = x']\right| \leq 0.4.$

Proof. Notice that $\Pr_a[X' = x']$ and $\Pr_b[X' = x']$ only depend on $|x'|_1$. So, the left-hand side of the inequality in the lemma is exactly

$$\sum_{j=0}^{N} \left|\Pr_a[|X'|_1 = j] - \Pr_b[|X'|_1 = j]\right| = \sum_{j=0}^{N} \binom{N}{j} \left|a^j(1-a)^{N-j} - b^j(1-b)^{N-j}\right|.$$

Notice that $\frac{a^j(1-a)^{N-j}}{b^j(1-b)^{N-j}} = \left(\frac{a}{b}\right)^j \left(\frac{1-a}{1-b}\right)^{N-j}$ is a decreasing function of j. Then let $j_1 \in [0, N]$ be the *real number* such that the ratio is exactly 1 when $j = j_1$. Then, for $j \in [0, j_1]$, we have $a^j(1-a)^{N-j} \geq b^j(1-b)^{N-j}$, and for every $j \in (j_1, N]$, we have $a^j(1-a)^{N-j} < b^j(1-b)^{N-j}$.

We can prove that $j_1 \in (Na, Nb)$. The following is a simple fact: Let $q \in (0, 1)$ be fixed, and x be a variable in $[0, 1]$, then $x^q(1-x)^{1-q}$ is maximized when $x = q$. (Consider the natural logarithm of $x^q(1-x)^{1-q}$, which is $q \ln x + (1-q)\ln(1-x)$. The derivative of the function is $\frac{q}{x} - \frac{1-q}{1-x}$, which is a decreasing function of x in the domain $(0, 1)$ and attains 0 value at $x = q$.) Therefore,

$$\left(\frac{a}{b}\right)^{Na}\left(\frac{1-a}{1-b}\right)^{N(1-a)} = \left(\frac{a^a(1-a)^{1-a}}{b^a(1-b)^{1-a}}\right)^N > 1, \qquad \left(\frac{a}{b}\right)^{Nb}\left(\frac{1-a}{1-b}\right)^{N(1-b)} = \left(\frac{a^b(1-a)^{1-b}}{b^b(1-b)^{1-b}}\right)^N < 1.$$

So, we have the left hand side of the inequality in the lemma is

$$2\sum_{j \leq j_1} \binom{N}{j}\left(a^j(1-a)^{N-j} - b^j(1-b)^{N-j}\right). \tag{4}$$

Using the Chernoff bound, we have

$$\Pr_a\left[|X'|_1 < \left(1 - \frac{4}{\sqrt{Na}}\right) Na\right] < e^{-\frac{8Na}{Na}} = e^{-8}.$$

This implies that the contribution of integers j in $[0, j_0)$ to (4) is at most $2e^{-8} \leq 0.01$.

Let $j_0 = Na - 4\sqrt{Na}$. Then, we only need to bound the contribution of integers j in $[j_0, j_1]$. This is done by proving that for every $j \in [j_0, j_1]$, we have

$$\frac{b^j(1-b)^{N-j}}{a^j(1-a)^{N-j}} > 0.9. \tag{5}$$

If this holds, then

$$\sum_{j\in[j_0,j_1]} \binom{N}{j} \left(a^j(1-a)^{N-j} - b^j(1-b)^{N-j}\right) \leq 0.1 \sum_{j\in[j_0,j_1]} \binom{N}{j} a^j(1-a)^{N-j} \leq 0.1.$$

So the contribution of integers j in $[j_0, j_1]$ to (4) is at most 0.2. This proves the lemma.

It remains to prove (5). Let $\theta = (j - Na)/\sqrt{N}$. Notice that θ may be positive or negative, but $|\theta| \leq \max\{(j_1 - Na)/\sqrt{N}, (Na - j_0)/\sqrt{N}\}$. Notice that $\frac{j_1 - Na}{\sqrt{N}} \leq \frac{N(b-a)}{\sqrt{N}} = \frac{c}{e^\epsilon+1}$ and $\frac{Na-j_0}{\sqrt{N}} = 4\sqrt{a} = \frac{4}{\sqrt{e^\epsilon+1}}$, When c is sufficiently small, the second upper bound is bigger, and thus $|\theta| \leq \frac{4}{\sqrt{e+1}} \leq 3$.

We bound the logarithm of the left side of (5):

$$\begin{aligned}
&j \cdot \ln\left(1 + \frac{b-a}{a}\right) + (N-j)\ln\left(1 - \frac{b-a}{1-a}\right) \\
&= \left(Na + \theta\sqrt{N}\right) \cdot \ln\left(1 + \frac{c}{\sqrt{N}}\right) + \left(N(1-a) - \theta\sqrt{N}\right)\ln\left(1 - \frac{e^\epsilon c}{\sqrt{N}}\right) \\
&\geq \left(Na + \theta\sqrt{N}\right)\left(\frac{c}{\sqrt{N}} - \frac{1}{2}\left(\frac{c}{\sqrt{N}}\right)^2\right) + \left(N(1-a) - \theta\sqrt{N}\right)\left(-\frac{e^\epsilon c}{\sqrt{N}} - \left(\frac{e^\epsilon c}{\sqrt{N}}\right)^2\right) \\
&= (1 + e^\epsilon)\theta c - \frac{c^2 a}{2} - (1-a)e^{2\epsilon}c^2 + \frac{\theta c^2}{\sqrt{N}}\left(e^{2\epsilon} - \frac{1}{2}\right).
\end{aligned}$$

Notice that $|\theta| \leq 3$. If we make c to be small enough constant, then the above quantity is at least -0.1, implying the left side of (5) is at least $e^{-0.1} \geq 0.9$. ☐

Lemma 24. There are probability measurement functions P'_0 and P'_1 such that

$$\begin{aligned}
&P_0(S) = (1-a)P'_0(S) + aP'_1(S) \\
&= P'_0(S) + a(P'_1(S) - P'_0(S)), \\
\text{and} \quad &P_0(S) + p\cdot(P_1(S) - P_0(S)) \\
&= (1-b)P'_0(S) + bP'_1(S) \\
&= P'_0(S) + b(P'_1(S) - P'_0(S)).
\end{aligned}$$

Proof. To guarantee the two equalities, we need

$$P'_0(S) = P_0(S) - \frac{ap(P_1(S) - P_0(S))}{b-a}, \qquad P'_1(S) = P_0(S) + \frac{(1-a)p}{b-a}(P_1(S) - P_0(S)).$$

It is easy to see that for every measurable S and its complement $\bar{S}$, we have $P'_0(S)+P'_0(\bar{S}) = 1$ and $P'_1(S)+P'_1(\bar{S}) = 1$. To show that P'_0 and P'_1 are probability measurement functions, we only need to prove that they are non-negative. Expressing a, b in terms of ϵ, we have any measurable set S,

$$\begin{aligned}
P'_0(S) &= P_0(S) - \frac{p(P_1(S) - P_0(S))}{(e^\epsilon - 1)p} = \frac{e^\epsilon \cdot P_0(S)}{e^\epsilon - 1} - \frac{P_1(S)}{e^\epsilon - 1} \geq 0, \\
P'_1(S) &= P_0(S) + \frac{e^\epsilon p(P_1(S) - P_0(S))}{(e^\epsilon - 1)p} = \frac{e^\epsilon \cdot P_1(S)}{e^\epsilon - 1} - \frac{P_0(S)}{e^\epsilon - 1} \geq 0.
\end{aligned}$$

☐

B $O(\frac{1}{\sqrt{\epsilon}})$-DP Algorithm for HST in Central Model

In this section we give our $O(\frac{1}{\sqrt{\epsilon}})$-approximate ϵ-DP algorithm for facility location under HST metrics, in the super-set output setting. This fixes a bug to the upper bound in Esencayi et al. (2019). Secondly, unlike Esencayi et al., our algorithm works for the case of non-uniform facility costs. The detail description is given in Algorithm 3.

In the algorithm we use P_v for every vertex $v \in V_T$ to denote the set of ancestors of v, including v itself. Let X denote the union of expensive vertices and minimal-cheap vertices. We set $\rho = \rho' = \frac{1}{\sqrt{\epsilon}}$ in the base algorithm. We still have $\tau \in \{\frac{1}{2}, 1, 2\}$. Let $\lambda = 2$ and $\eta = \sqrt{\lambda}$.

Recall that a vertex v is *cheap* if $\lambda^{\ell(v)} \geq \sqrt{\epsilon} \cdot f_v$, otherwise *expensive*. We say a vertex v is *minimal-cheap* if its children are all expensive.

Algorithm 3 DP-FL-tree(ϵ)

1: **for** every v in X, define $\tilde{N}_v := N_v + \mathrm{Lap}\left(\frac{\sqrt{f_v}}{c \cdot \epsilon^{3/4} \cdot \eta^{\ell(v)}}\right)$, where $c = \frac{\eta-1}{\eta^3}$.
2: let $M' \leftarrow \left\{v \in V_T : \lambda^{\ell(v)} \geq \sqrt{\epsilon} f_v \text{ or } \tilde{N}_v \cdot \lambda^{\ell(v)} \geq f_v/\sqrt{\epsilon}\right\}$ ▷ vertices in M' will be called *noisily-marked* vertices, and other vertices are said to be *noisily-unmarked*.
3: $C \leftarrow \left\{v \in M' : \forall u \in ((P_v \setminus \{v\}) \cap X), \tilde{N}_u \cdot \lambda^{\ell(u)} \geq f_v/\sqrt{\epsilon})\right\}$
4: $R' \leftarrow \text{min-set}(C)$
5: **return** R' but only open $S' := \text{open}(R')$

The main difference between the base algorithm with $\tau = 1$ and here (other than that between $\tilde{N}_v$'s and N_v's) is that we introduce a new filtering operation to obtain the set C in Step 3, and our R' is open(C). It is easy to see that any node v with $\lambda^{\ell(v)} \geq f_v\sqrt{\epsilon}$ is in C.

By the way we define the noise for vertices, we can show that the algorithm is ϵ-DP:

Lemma 25. Algorithm 3 satisfies ϵ-DP.

Proof. Consider two neighboring data sets D and D', and let v be the unique leaf vertex that the two data sets differ. We will prove ϵ-DP of set M', and that would be sufficient since set R' is completely decided by M'.

First of all, for all $u \in V_T$ which is not an ancestor of v, $\tilde{N}_u|D$ and $\tilde{N}_u|D'$ have the same distribution. We also do not need to worry for u such that $\lambda^{\ell(u)} \geq \sqrt{\epsilon} f_u$, because they are always marked. So, we only need to look into the case $\lambda^{\ell(u)} < \sqrt{\epsilon} f_u$ and u is an ancestor of v.

Due to the property of Laplacian distribution, the sub-algorithm for such a u is $\frac{c\epsilon^{3/4}\eta^{\ell(u)}}{\sqrt{f_u}}$-differentially private. Note that we are only interested in vertices u such that $\frac{\lambda^{\ell(u)}}{f_u} < \sqrt{\epsilon}$, or equivalently, $\frac{\eta^{\ell(u)}}{\sqrt{f_u}} < \epsilon^{1/4}$. Let $\ell' = \ell(u')$ where u' is the maximum level u satisfying this condition. Now, let us figure out the privacy budget we are using for all vertices of interest, starting from u' going down towards v. For u', the privacy budget we are using is $\frac{c\epsilon^{3/4}\eta^{\ell(u')}}{\sqrt{f'_u}} < c\epsilon^{3/4}\epsilon^{1/4} = c\epsilon$. For the next vertex u (child of u'), we're spending $\frac{c\epsilon^{3/4}\eta^{\ell(u)}}{\sqrt{f_u}} = \frac{c\epsilon^{3/4}\eta^{\ell'-1}}{\sqrt{f_u}} < \frac{c\epsilon^{3/4}}{\eta} \cdot \frac{\eta^{\ell'}}{\sqrt{f'_u}} \leq \frac{c\epsilon^{3/4}}{\eta}\epsilon^{1/4} = \frac{c}{\eta} \cdot \epsilon$. Similarly, the privacy budgets we're spending are at most $c\epsilon$, $\frac{c}{\eta}\epsilon$, $\frac{c}{\eta^2}\epsilon$, and so on, starting from u' towards v. If we add them all for all such u vertices, we have

$$c\epsilon + \frac{c}{\eta}\epsilon + \frac{c}{\eta^2}\epsilon + \cdots + \frac{c}{\eta^{\ell'}}\epsilon = c\epsilon\sum_{t=0}^{\ell'}(\frac{1}{\eta})^t < c\epsilon\frac{1}{1-\frac{1}{\eta}} = c\epsilon\frac{\eta}{\eta-1} = \epsilon.$$ □

The following two lemmas show that the extra facility and connection cost incurred by the noise is small in expectation:

Lemma 26. The cost of facilities in S' is $O(\frac{1}{\sqrt{\epsilon}}) \cdot \mathrm{opt}$.

Proof. We break S' into two parts: $S' \cap M$ and $S' \setminus M$. By Lemma 18, the cost of $S' \cap M$ is at most $O(\frac{1}{\sqrt{\epsilon}}) \cdot \text{opt}$. Therefore, it suffices for us to bound the cost of $S' \setminus M \subseteq M' \setminus M$, i.e, the unmarked but noisily-marked facilities. Focus on a vertex $v \notin M$; that is, a vertex v satisfying $\lambda^{\ell(v)} < \sqrt{\epsilon} f_v$ and $N_v \lambda^{\ell(v)} < \frac{f_v}{2\sqrt{\epsilon}}$. We bound the probability that $v \in M'$, i.e, $\tilde{N}_v \lambda^{\ell(v)} \geq \frac{f_v}{\sqrt{\epsilon}}$, using the following property of Laplace distribution:

Lemma 27. If $Y \sim \mathit{Laplace}(b)$, then $\Pr[|Y| \geq tb] = \exp(-t)$ for any t.

With the lemma, we can bound the probability:

$$\begin{aligned}\Pr[v \in M'] &= \Pr\left[\tilde{N}_v \geq \frac{f_v}{\lambda^{\ell(v)}\sqrt{\epsilon}}\right] \\ &\leq \Pr\left[\tilde{N}_v \geq N_v + \frac{f_v}{2\lambda^{\ell(v)}\sqrt{\epsilon}}\right] = \Pr\left[\tilde{N}_v - N_v \geq \frac{f_v}{2\lambda^{\ell(v)}\sqrt{\epsilon}}\right] \\ &= \exp\left(-\frac{f_v}{2\lambda^{\ell(v)}\sqrt{\epsilon}} \Big/ \frac{\sqrt{f_v}}{c\epsilon^{3/4}\eta^{\ell(v)}}\right) = \exp\left(-\frac{c}{2}\epsilon^{1/4}\sqrt{\frac{f_v}{\lambda^{\ell(v)}}}\right).\end{aligned}$$

Note that $\tilde{N}_v - N_v$ is the Laplacian noise we are adding in the algorithm and we used Lemma 27. Thus, we have

$$\Pr[v \in M'] \cdot f_v \leq f_v \cdot \exp\left(-\frac{c}{2}\epsilon^{1/4}\sqrt{\frac{f_v}{\lambda^{\ell(v)}}}\right) = \frac{1}{\sqrt{\epsilon}} \cdot \lambda^{\ell(v)} \cdot y^2 \cdot \exp\left(-\frac{c}{2} \cdot y\right),$$

where $y = \epsilon^{1/4} \cdot \sqrt{\frac{f_v}{\lambda^{\ell(v)}}}$. One can easily show that the function $g(y) = y^2 \cdot \exp(-\frac{c}{2} \cdot y)$ is bounded by a constant $d = \frac{16}{c^2} \cdot e^{-2}$. Now, we have

$$\Pr[v \in M'] \cdot f_v \leq \frac{d}{\sqrt{\epsilon}} \cdot \lambda^{\ell(v)} \tag{6}$$

The vertices of $S' \setminus M$ are divided into two cases below.

1. The vertices $v \in S' \setminus M$ with $N_v \geq 1$. Let S'_1 denote that set. Let $T_1(v) := \{u | u \in T_v \setminus M, N_u \geq 1\}$. The expected open facility cost in $T_1(v)$ is at most

$$\mathbf{Cost}_1(T_v) = \sum_{u \in T_1(v)} \Pr[u \in M' | u \notin M] \cdot f_u \leq \sum_{u \in T_1(v)} \frac{d}{\sqrt{\epsilon}} \cdot \lambda^{\ell(u)} = O(1) \cdot \frac{1}{\sqrt{\epsilon}} \cdot \min\{N_v \lambda^{\ell(v)}, f_v\}$$

   The facility open cost of S'_1 is bounded by

$$\mathbf{Cost}_1 \leq \sum_{v \in V_1} \mathbf{Cost}_1(T_v) \leq O(1/\sqrt{\epsilon}) \sum_{v \in V_1} B(v) = O(\text{opt}/\sqrt{\epsilon}),$$

   where $V_1 := \{u \notin M, (P_u \setminus \{u\}) \subseteq M, N_u \geq 1\}$ is a set of maximal-expensive points each with positive number of demand clients, and $\text{opt} \geq \sum_{v \in V_1} B(v)$ based on Corollary 11 since V_1 does not have any ancestor-descendant pair.

2. The vertices u with $N_u = 0$ in $S' \setminus M$. Let S'_0 denote that set of vertices. For each $v \in X$, let $T_0(v) := \{u : u \in T_z, P(z) = v, N_z = 0\}$ denote the union of nodes of subtrees each with parent be v and has zero demand points, recall that $P(u)$ denotes the parent of u. Let $Z_v/\sqrt{\epsilon} = \tilde{N}_v \cdot \lambda^{\ell(v)}$ if $\tilde{N}_v \cdot \lambda^{\ell(v)} \geq f_v/\sqrt{\epsilon}$ and $Z_v = 0$ otherwise, where $v \in X$. Let us denote the expectation of Z_v as $E[Z_v]$, which is

$$\begin{aligned}E[Z_v] &= \sqrt{\epsilon} \cdot \lambda^{\ell(v)} \left(N_v + \int_{\frac{f_v}{(\lambda^{\ell(v)}\sqrt{\epsilon})} - N_v}^{\infty} \frac{x}{2b} exp(-\frac{|x|}{b}) dx\right) \\ &\leq \sqrt{\epsilon} \cdot \lambda^{\ell(v)} \left(2N_v + \int_{\frac{f_v}{(\lambda^{\ell(v)}\sqrt{\epsilon})}}^{\infty} \frac{x}{2b} exp(-\frac{x}{b}) dx\right) = O(1) \cdot \frac{\max\{N_v, 1\}}{\sqrt{\epsilon}} \cdot \lambda^{\ell(v)}\end{aligned} \tag{7}$$

when $b = \left(\frac{\sqrt{f_v}}{c\cdot\epsilon^{3/4}\cdot\eta^{\ell(v)}}\right) > 0$ (the conclusion above is trivially true for the case $b = 0$) and $\frac{1}{2b}exp(-\frac{|x|}{b})$ is the density function of $Laplace(b)$. We use indefinite integral that $\int \frac{x}{2b}exp(-\frac{x}{b})dx = -\frac{1}{2}exp(-\frac{x}{b})(b+x)$ in the computation above. Let $\frac{f_v}{(\lambda^{\ell(v)}\sqrt{\epsilon})} = \gamma\cdot b$ where $\gamma \geq 0$. We have $\int_{\frac{f_v}{(\lambda^{\ell(v)}\sqrt{\epsilon})}}^{\infty} \frac{x}{2b}exp(-\frac{x}{b})dx = \frac{1}{2}exp(-\gamma)(\gamma+1)b$. We have $exp(-\gamma)\cdot\gamma b = \exp\left(-\epsilon^{1/4}\sqrt{\frac{f_v}{\lambda^{\ell(v)}}}\right)\frac{f_v}{(\lambda^{\ell(v)}\sqrt{\epsilon})} = \frac{O(1)}{\epsilon}$ based on formula before (6). Also we have $exp(-\gamma)\cdot b = \exp\left(-\epsilon^{1/4}\sqrt{\frac{f_v}{\lambda^{\ell(v)}}}\right)\cdot\left(\frac{\sqrt{f_v}}{c\cdot\epsilon^{3/4}\cdot\eta^{\ell(v)}}\right) = \exp\left(-\epsilon^{1/4}\sqrt{\frac{f_v}{\lambda^{\ell(v)}}}\right)\cdot\left(\epsilon^{1/4}\sqrt{\frac{f_v}{\lambda^{\ell(v)}}}\right)\cdot\frac{1}{c\epsilon} = \frac{O(1)}{\epsilon}$.

Note that $|T_0(v)\cap S'| > 1$ could not happen based on genetically closest facility assignment rule. For each $v \in X$, we denote $\mathbf{Cost}(T_0(v))$ as the total facility open cost of nodes in $T_0(v)$, and its expectation $E[\mathbf{Cost}(T_0(v))]$ is bounded by $E[Z_v]$, because we add the facility $w \in T_0(v)$ with facility cost f_w ($f_w \geq f_v$) to C only if $\tilde{N}_v \cdot \lambda^{\ell(v)} \geq f_w/\sqrt{\epsilon}$. Define $\mathbf{C}_v$ the as condition that $S' \cap (T_v \setminus T_0(v)) = \emptyset$ where $v \in X$. Note that adding condition C_v would not increase $E[Z_v]$.

1). If condition $\mathbf{C}_v$ does not hold on, then $|T_0(v)\cap S'| = 0$ based on genetically closest facility assignment rule. In that case we have the number of open facilities in $T_0(v)$ is zero and $\mathbf{Cost}(T_0(v)) = 0$.

2). If condition $\mathbf{C}_v$ hold on, then sum connection cost of nodes in T_v is at least $N_v \cdot \lambda^{\ell(v)}$. That means $E[\mathbf{Cost}(T_0(v))]$ can be charged to connection cost of T_v by $O(1/\sqrt{\epsilon})$ factor when $N_v \geq 1$. Note that for any ancestor-descendant pair (v, w) where C_v holds on, v is the ancestor of w and $w \notin T_0(v)$, then we have that the number of open facilities in $T_0(w)$ is zero.

Note that $S_0' \subseteq X$ as each of cheap vertex is in C. Also any point $u \in S_0'$ has at least one ancestor $v \in X$ with $N_v \geq 1$. The facility open cost of S_0' is bounded by $\mathbf{Cost}_2 \leq \sum_{v\in V_2}\mathbf{Cost}(T_0(v))$ where $V_2 := \{N_v \geq 1, C_v \text{ hold on }, v \in X : \forall u \in ((P_v \setminus \{v\}) \cap X), C_u \text{ does not hold on }\}$.

Hence $E[\mathbf{Cost}_2]$ can be charged to sum of connection cost (denote that sum as $\mathbf{Cost}_c$) within $O(1/\sqrt{\epsilon})$ factor, namely $O(\frac{1}{\sqrt{\epsilon}})\mathbf{Cost}_c$. Also the total connection cost in solution produced by Algorithm 3 is $O(\text{opt})$ (It was shown in Lemma 28). The facility open cost of S_0' is bounded by $O(\frac{1}{\sqrt{\epsilon}}\text{opt})$.

This finishes the proof of Lemma 26. □

Lemma 28. The expected increase of connection cost in Algorithm 3, is $O(1)$ times that of Algorithm 1 (main body of paper) with parameter $\rho = \rho' = \frac{1}{\sqrt{\epsilon}}$ and $\tau = 1$.

Proof. For the convenience of our analysis, we will assume that every client is connected to its lowest noisily marked ancestor. Notice that the actual costs may be larger by a factor of 2 and it can be ignored.

Focus on a vertex $v \in S$, so some clients are connected to v in the base algorithm. An additional connection cost in T_v will only incur in the case that v is not open in Algorithm 3 and vertices formerly connected to v will have to connect some ancestor of v. Let the ancestors of v(including itself) from the bottom to the top be $v_0 = v, v_1, v_2, \cdots$. Due to the symmetric property of laplace noise, $\Pr[v_0 \notin M'] \leq 1/2$ and in that case the connection cost increases by a factor of λ. We only need to consider the case $\lambda^{\ell(v)} \leq \sqrt{\epsilon}f_v$, otherwise $\lambda^{\ell(v)} > \sqrt{\epsilon}f_v$ and $v_i \notin X$. Hence we have either

$$\begin{aligned}\Pr[\tilde{N}_{v_i}\cdot\lambda^{\ell(v_i)} < f_v/\sqrt{\epsilon}] &\leq \Pr\left[\tilde{N}_{v_i} \leq \frac{f_v}{\sqrt{\epsilon}\lambda^i\lambda^{\ell(v)}}\right] = \Pr\left[\tilde{N}_{v_i} - N_{v_i} \leq -(1-1/\lambda^i)\frac{f_v}{\sqrt{\epsilon}\lambda^{\ell(v)}}\right]\\ &= \exp\left(-(1-1/\lambda^i)\frac{\frac{f_v}{\sqrt{\epsilon}\lambda^{\ell(v)}}}{\frac{\sqrt{f_v}}{c\epsilon^{3/4}\eta^{\ell(v)-i}}}\right) = \exp\left(-(1-1/\lambda^i)\eta^i c\epsilon^{1/4}\sqrt{\frac{f_v}{\lambda^{\ell(v)}}}\right)\\ &\leq \exp\left(-(1-1/\lambda)c'\eta^i\right)\end{aligned}$$

or $v_i \notin X$. The probability that $v_{i-1} \notin C$ ($i > 0$) assuming its ancestors are subset of C, is at most $\exp(-c'\eta^i)$ where $c' = -(1-1/\lambda)c$. The general that $v_i \notin C$ the connection cost increases by a factor of and the corresponding increase of connection cost is by a factor of λ^i. Therefore, the expected scaling factor for the connection cost dues to the noise is at most

$$1/2 + \sum_{i=1}^{\infty}(\lambda)^i\exp(-c'\lambda^{i/2}) = O(1).$$ □

Combining the Lemma 26 and Lemma 28 gives the result.

Theorem 29. Algorithm 3 gives $O(\frac{1}{\sqrt{\epsilon}})$-approximation.

Finally, our connection cost is only $O(1) \cdot \mathrm{opt}$. So, in the general metric, the ϵ-DP algorithm gives an $O(\log n)$-approximation, assuming ϵ is not too small.