A $(4+\epsilon)$-Approximation for Euclidean k-Means via Non-Monotone Dual-Fitting

Moses Charikar
Stanford University
Stanford, USA
moses@cs.stanford.edu

Vincent Cohen-Addad
Google Research
New York, USA
cohenaddad@google.com

Ruiquan Gao
Stanford University
Stanford, USA
ruiquan@cs.stanford.edu

Fabrizio Grandoni
IDSIA, USI-SUPSI
Lugano, Switzerland
fabrizio.grandoni@gmail.com

Euiwoong Lee
University of Michigan
Ann Arbor, USA
euiwoong@umich.edu

Ernest van Wijland
Université Paris-Cité, CNRS
Paris, France
ernest.vanwijland@irif.fr

Abstract

We present a polynomial-time $(4+\epsilon)$-approximation algorithm for (high-dimensional) Euclidean k-Means. This substantially improves on the current-best 5.83-approximation in [Charikar, Cohen-Addad, Gao, Grandoni, Lee, Van Wijland - FOCS'25] (that also works for the metric case).

The mentioned algorithm by Charikar et al. critically exploits a greedy Lagrangian Multiplier Preserving (LMP) approximation for Facility Location with squared metric distances, that adapts the classical greedy algorithm with dual-fitting analysis for Metric Facility Location in [Jain, Mahdian, Markakis, Saberi, Vazirani - J.ACM'03]. The authors then turn it into an approximation algorithm for (Metric) k-Means, at the cost on an extra factor $1+\epsilon$, by exploiting the framework introduced in [Cohen-Addad, Grandoni, Lee, Schwiegelshohn, Svensson - STOC'25] for k-Median.

Our main contribution is a greedy LMP 4-approximation for Facility Location with squared *Euclidean* distances. Differently from Charikar et al., our algorithm sometimes decreases the dual variables, a quite uncommon feature for dual-based algorithms. This is critical in our dual-fitting analysis in order to exploit the specific properties of Euclidean metrics. For the $(4+\epsilon)$-approximation for k-Means, we extend the framework by Cohen-Addad et al. by overcoming substantial technical challenges posed by decreased dual values.

CCS Concepts

• **Theory of computation → Facility location and clustering**.

Keywords

Approximation Algorithms, Clustering, k-Means

ACM Reference Format:
Moses Charikar, Vincent Cohen-Addad, Ruiquan Gao, Fabrizio Grandoni, Euiwoong Lee, and Ernest van Wijland. 2026. A $(4+\epsilon)$-Approximation for Euclidean k-Means via Non-Monotone Dual-Fitting. In *Proceedings of the 58th Annual ACM Symposium on Theory of Computing (STOC '26), June 22–26, 2026, Salt Lake City, UT, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3798129.3800894

This work is licensed under a Creative Commons Attribution 4.0 International License.
STOC '26, Salt Lake City, UT, USA
© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-2536-4/2026/06
https://doi.org/10.1145/3798129.3800894

1 Introduction

Euclidean k-Means is arguably the most well-studied clustering problem in Euclidean spaces. Given a set $D \subseteq \mathbb{R}^t$ and $k \in \mathbb{N}$, the goal is to find k centers $S = \{s_1, \ldots, s_k\} \subseteq \mathbb{R}^t$ to minimize

$$\sum_{j\in D} d^2(j,S) = \sum_{j\in D} \min_{i\in S} d^2(j,i),$$

where $d^2(x,y) := \|x-y\|_2^2$ denotes the squared Euclidean distance. It has been studied at least as early as the 1950s [31, 32, 36] with wide applications to signal/image processing, data mining, statistics, and machine learning.

The best-known approximation factor is a 5.83-approximation by Charikar, Cohen-Addad, Gao, Grandoni, Lee, and Van Wijland [6]. This improves on a sequence of increasingly better approximations for the problem: a $(9+\epsilon)$-approximation by Kanungo, Mount, Netanyahu, Piatko, Silverman, and Wu [25], a 6.36-approximation by Ahmadian, Norouzi-Fard, Svensson, and Ward [1], a 6.13-approximation by Grandoni, Ostrovsky, Rabani, Schulman, and Venkat [19], and a 5.92-approximation by Cohen-Addad, Esfandiari, Mirrokni, and Narayanan [10].

However, there is still a wide gap between algorithms and hardness results. Even the NP-hardness of the exact optimization was proved less than 20 years ago [15]. And after a series of papers [3, 14, 27], the best hardness of approximation factor is 1.06 assuming P $\neq$ NP and 1.36 assuming the Johnson Coverage Hypothesis [14].

Euclidean k-Means can be further generalized to Metric k-Means, where a general metric space $(D \cup \mathcal{F}, d)$ is explicitly given and the goal is to find k centers $S = \{s_1, \ldots, s_k\} \subseteq \mathcal{F}$ to minimize $\sum_{j\in D} d^2(j,S)$. Given an instance $D \subseteq \mathbb{R}^t$ and k for the Euclidean version, known reductions [16, 17, 34] allow one to focus on a set of candidate centers $\mathcal{F}$ of size $n^{O_\epsilon(1)}$ while introducing a factor $1+\epsilon$ in the approximation for any constant $\epsilon > 0$. Therefore, one can (essentially) see Metric k-Means as a generalization of Euclidean k-Means. The aforementioned 5.83-approximation for the Euclidean version [6] indeed holds for the general metric version, while the hardness of approximation factor is much larger for the general metric case compared to the Euclidean case, namely $1 + 8/e \approx 3.94$ [23].

Our main result is a $(4+\epsilon)$-approximation for Euclidean k-Means for any constant $\epsilon > 0$, which significantly improves the 5.83-approximation in [6] and gets close to the 3.94-hardness for general metrics.

Theorem 1. *For any sufficiently small constant $\epsilon > 0$, there is a randomized polynomial-time $(4+\epsilon)$-approximation for Euclidean k-Means.*

1.1 Related Work

k-Means, k-Median, and k-Center are three of the most well-studied clustering problems in a metric space where the target number of clusters k is fixed. Given a metric space $(D \cup \mathcal{F}, d)$, in k-Center, the goal is to choose a set $S \subseteq \mathcal{F}$ of k centers in order to minimize $\max_{j\in D} d(j,S)$. k-Center admits a simple greedy 3-approximation[1].

k-Median is defined like k-Means, except that the goal is to minimize the sum of the distances rather than squared distances, i.e., the objective function is $\sum_{j\in D} d(j,S)$. k-Median is very close to k-Means in terms of results and techniques. k-Median is hard to approximate below a factor $1+2/e$ [23]. For general metric distances, the first constant approximation was achieved by Charikar, Guha, Tardos and Shmoys [8]. After a very long sequence of improvements [2, 5, 12, 18, 22, 24, 30], the current-best $(2+\epsilon)$-approximation for this problem was very recently achieved by Cohen-Addad, Grandoni, Lee, Schwiegelshohn, and Svensson [13]. This is also the best result for the Euclidean case, improving on an earlier 2.406-approximation [10].

Another closely related problem is Uncapacitated Facility Location, where given a metric space $(D \cup \mathcal{F}, d)$ and facility opening costs $f : \mathcal{F} \to \mathbb{R}^+$, the goal is to open a subset of facilities $S \subseteq \mathcal{F}$ (without cardinality restriction) to minimize $\sum_{s\in S} f(s) + \sum_{j\in D} d(j,S)$. For general metrics, the best-known approximation factor is 1.488 [29], which is a culmination of another long line of work [4, 7, 9, 20, 22, 24, 33, 35]. This is very close to the best-known lower bound [20] 1.463. A slightly better approximation is known for the Euclidean version [28]. The greedy algorithm by Jain, Mahdian, Markakis, Saberi, and Vazirani [22] gives an LMP 2-approximation for this problem. The LMP approximation factor was recently improved slightly below 2 by Cohen-Addad, Grandoni, Lee and Schwiegelshohn [12].

Euclidean k-Means with a small value of k or a small dimension t has been actively studied as well. When both k and t are constants, the problem can be solved exactly in polynomial time [21]. Furthermore, there are $(1+\epsilon)$-approximation algorithms that run in time $f(\epsilon,k)\mathrm{poly}(n)$ [17, 26] and $f(\epsilon,t)\mathrm{poly}(n)$ [11].

2 Overview of our Approach

At a very high level, our approach follows the general two-step framework that yielded a recent $(2+\epsilon)$-approximation for Metric k-Median [13] and a 5.83-approximation for Metric k-Means [6]. The framework is a combination of two different algorithms. The first one is a bicriteria approximation algorithm with the desired approximation factor, which opens $O(\log n/\epsilon^3)$ more centers than the k allowed ones. Let opt denote the optimal solution cost for the considered instance.

[1]For the special case $D = \mathcal{F}$, there is a 2 approximation which is also tight.

Theorem 2. *For any sufficiently small constant $\epsilon > 0$, there is a polynomial-time algorithm for Euclidean k-Means that returns a solution containing at most $k + O(\log n/\epsilon^3)$ centers and of cost at most $(4+O(\sqrt{\epsilon}))opt$.*

The second algorithm is a $(4+O(\sqrt{\epsilon}))$-approximation algorithm that only works for instances which are *stable* in the following sense. We say that an instance of k-Means is β-stable if $\mathrm{opt}_{k-1} \geq (1+\beta)\mathrm{opt}$, where $\mathrm{opt}_{k'}$ is the optimal cost for the same instance but with the target number of centers being k'. This algorithm works for the more general metric case, however under the assumption that $D \subseteq \mathcal{F}$.

Theorem 3. *For any constants $\epsilon, \zeta > 0$, there exists a randomized polynomial-time algorithm that, given a $(\zeta/\log n)$-stable Metric k-Means instance with $D \subseteq \mathcal{F}$, returns a solution of cost at most $(4+O(\sqrt{\epsilon}))opt$ with high probability.*

It is relatively easy to derive Theorem 1 from the above two theorems.

Proof of Theorem 1. We compute a set of feasible solutions, and return the cheapest one. Let Δ be the maximum extra number of centers computed by the algorithm from Theorem 2 (this number is independent from k). Let $k' = \max\{1, k-\Delta\}$. One solution is obtained by computing the optimum solution with one center if $k' = 1$, and otherwise by running the algorithm from Theorem 2 with target number of centers being k' (notice that it returns a solution with at most k centers, hence feasible). Furthermore, we apply the reduction in [16, 17, 34] to obtain an instance of Metric k-Means with candidate centers $\mathcal{F}$. W.l.o.g. we can enforce that $D \subseteq \mathcal{F}$ since we can open centers at any location. Let $\overline{\mathrm{opt}}_{k'}$ indicate the optimal cost for the latter instance with k' centers. Observe that $\mathrm{opt}_{k'} \leq \overline{\mathrm{opt}}_{k'} \leq (1+\epsilon)\mathrm{opt}_{k'}$. We run the algorithm from Theorem 3 for the obtained metric instance for every integer $k'' \in (k', k]$ (thus obtaining solutions with $k'' \leq k$ centers, hence feasible).

If $\mathrm{opt}_{k'} \leq (1+\epsilon)^2\mathrm{opt}_k$, the first solution is $(1+\epsilon)^2(4+O(\sqrt{\epsilon}))$ approximate. Otherwise, observe that $\overline{\mathrm{opt}}_{k'} \geq \mathrm{opt}_{k'} \geq (1+\epsilon)^2\mathrm{opt}_k \geq (1+\epsilon)\overline{\mathrm{opt}}_k$. Thus there exists an integer $k'' \in (k', k]$ such that $\overline{\mathrm{opt}}_{k''} \leq (1+\epsilon)\overline{\mathrm{opt}}_k$ and $\overline{\mathrm{opt}}_{k''-1} \geq (1+\beta)\overline{\mathrm{opt}}_{k''}$ for $\beta \in \Omega(\epsilon/\Delta) = \Omega(\epsilon^4/\log n)$. For that value of k'' the corresponding instance of Metric k-Means is β-stable, hence the respective solution has cost at most

$$(4+O(\sqrt{\epsilon}))\overline{\mathrm{opt}}_{k''} \leq (1+\epsilon)(4+O(\sqrt{\epsilon}))\overline{\mathrm{opt}}_k \leq (1+\epsilon)^2(4+O(\sqrt{\epsilon}))\mathrm{opt}_k$$

The claim follows by rescaling ϵ. □

It remains to describe how the above two theorems are obtained. The algorithm from Theorem 3 is an adaptation of a rather complex $5+O(\sqrt{\epsilon})$ approximation for stable instances of Metric k-Means in [13]. The assumption $D \subseteq \mathcal{F}$, that is w.l.o.g. for metric instances arising from Euclidean ones, leads to a massive simplification of their algorithm and analysis, while improving the approximation factor to $4+O(\sqrt{\epsilon})$.

The main contribution of this paper is our proof of Theorem 2. In more detail, we consider the (Uncapacitated) Facility Location problem with squared Euclidean distances and uniform opening costs. The input is $D \subseteq \mathbb{R}^t$ and the uniform facility cost f. We are allowed to open an arbitrary number of centers/facilities $S \subseteq \mathbb{R}^t$, and the objective function is the total cost of the open facilities plus the squared distance from each client to the closest (open) facility in S, i.e.,

$$\mathrm{cost}_{FL}(S) := f|S| + \sum_{j\in D} d^2(j,S)\,.$$

Using the reductions in [16, 17, 34], in polynomial time we construct a set $\mathcal{F}$ of candidate centers for the clusters so that, if we restrict to those centers, the cost of the solution grows at most by a factor $(1+\epsilon)$. Based on that we define the following LP relaxation for the problem and its dual:

$$\begin{aligned} \min \quad & \sum_{i\in\mathcal{F}, j\in D} d^2(i,j)x_{i,j} + f\cdot\sum_{i\in\mathcal{F}} y_i && (LP_{FL}(f)) \\ \text{s.t.} \quad & \sum_{i\in\mathcal{F}} x_{i,j} \geq 1 && \forall j\in D \\ & y_i - x_{i,j} \geq 0 && \forall j\in D, i\in\mathcal{F} \\ & x,y\geq 0. \end{aligned}$$

$$\begin{aligned} \max \quad & \sum_{j\in D} \alpha_j && (DP_{FL}(f)) \\ \text{s.t.} \quad & \sum_{j\in D} [\alpha_j - d^2(i,j)]^+ \leq f && \forall i\in\mathcal{F} \\ & \alpha\geq 0. \end{aligned}$$

Above, $[a]^+ := \max(a,0)$. Let $\mathrm{opt}_{LP}(f)$ be the optimal values for $LP_{FL}(f)$.

We say that an algorithm for the above problem is Lagrangian Multiplier Preserving (LMP) Γ-approximate if it produces a feasible solution S such that

$$\Gamma\cdot f|S| + \sum_{j\in D} d^2(j,S) \leq \Gamma\cdot \mathrm{opt}_{LP}(f). \tag{1}$$

In other words, the solution costs at most Γ times the optimum even if we increase the facility cost of S by a factor Γ.

For Metric Facility Location, the two prominent ways to obtain an LMP approximation are the primal-dual method of Jain and Vazirani [24] and the greedy algorithm (with the dual-fitting analysis) of Jain, Mahdian, Markakis, Saberi, and Vazirani [22]. It is relatively easy to adapt the primal-dual algorithm to obtain an LMP 9-approximation for Facility Location with squared metric distances [1]. The absence of triangle inequality has made it hard to generalize the greedy algorithm too. A recent work [6] suggests a modification of the greedy algorithm, which bypasses this difficulty and yields an LMP 5.83-approximation. Our main technical contribution is a further refinement of this direction specifically exploiting the structure of Euclidean spaces, which results in an LMP 4-approximation for Facility Location with squared Euclidean distances.

The Greedy Algorithm in [6]. Let us review the greedy algorithm of [6]. The algorithm has a variable α_j per client j (initialized to 0), a set S of open facilities (initialized to $\emptyset$), and a set A of *active* clients (initialized to D). We partition the remaining *inactive* clients into the directly connected ones DC and indirectly connected ones IC. We choose some parameter $\gamma > 1$ and let $\Gamma = \gamma + \frac{2\gamma}{\gamma-1}$. We define a scaled facility cost $\hat{f} = \Gamma f$. At each point of time, the *bid* $bid(j,i)$ of client j towards facility i is $[\alpha_j - \gamma d(j,i)]^+$ if j is active or indirectly connected and $\gamma[d(j,S) - d(j,i)]^+$ otherwise[2]. The variables α_j of active clients are increased uniformly until one of the following events happens:

(a) For some client j and $i\in S$, $\alpha_j \geq d(j,i)$. In that case j is moved from A to IC, and we say that j is connected to i;

(b) For some (not open) facility $i\notin S$, one has $\sum_{j\in D} bid(j,i) = \hat{f}$. In that case we open i, i.e., we add i to S. Furthermore, each $j\in D$ with $bid(j,i) > 0$ is moved to DC, and (re)connected to j. Also, each client $j\in A$ with $\alpha_j \geq d^2(j,i)$ is moved to IC and connected to i.

Intuitively, active clients are not yet connected to any open facility, and their dual variables are actively growing. Directly connected clients are connected to an open facility and directly contribute a positive amount to the opening cost of that facility. On the other hand, indirectly connected clients are connected to an open facility but they don't contribute a positive amount to the opening cost of that facility.

A simple attempt for an LMP 4-approximation. In the analysis of [6] for Facility Location with squared metric distances, the authors use two main constraints to build up a factor-revealing LP given IC, DC. More specifically, consider any facility i and the set $D^* = \{j\in D : \alpha_j > \Gamma d^2(i,j)\}$. Let $D^* = \{1,\dots,t\}$, sorted in non-decreasing order of deactivation time α_j. We next restrict our attention only to the clients in D^*. Let IC^j denote the set of active or indirectly connected clients and DC^j denote the set of directly connected clients among the first $j-1$ clients right before j becomes inactive. Let also S^j denote the set of open facilities right before j becomes inactive. The two constraints are then $\forall j'\in DC^j$:

$$\alpha_j \leq d^2(j,S^j) \leq \gamma\cdot d^2(j',S^j) + \frac{2\gamma}{\gamma-1}\cdot(d^2(i,j)+d^2(i,j')), \tag{2}$$

and

$$\begin{aligned}\sum_{j'\in DC^j}(\gamma d^2(j',S^j) - \gamma d^2(j',i)) + \sum_{j'\in IC^j}(\alpha_{j'} - \gamma d^2(j',i)) \\ + \sum_{j'\geq j}(\alpha_j - \gamma d^2(j',i)) \leq \hat{f}\end{aligned} \tag{β_j}$$

The first constraint is based on an approximate form of triangle inequality for three-hop paths in squared metrics (i.e., $\gamma x^2 + \left(2+\frac{2}{\gamma-1}\right)(y^2+z^2) \geq (x+y+z)^2$ for any $\gamma > 1$ and x,y,z). The second one simply states that the clients do not *overbid* to open a facility.

A possible approach to improve on the analysis in [6] is to replace (2) with a variant for two-hop paths (i.e., $\gamma x^2 + \frac{\gamma}{\gamma-1}y^2 \geq (x+y)^2$ for any $\gamma > 1$ and x,y), which gives us smaller coefficients to improve the LMP approximation ratio:

$$\alpha_j \leq d^2(j,S^j) \leq \gamma\cdot d^2(j',S^j) + \frac{\gamma}{\gamma-1}\cdot d^2(j,j')\,. \tag{3}$$

This would allow us to replace $\Gamma = \gamma + \frac{2\gamma}{\gamma-1}$ with $\Gamma = \gamma + \frac{\gamma}{\gamma-1}$, which then gives the desired bound for $\gamma = 2$. In more detail, following the derivation of [6], if DC^j contains all the first $j-1$ clients in D^*, which is the worst case in the analysis of [6], we can easily obtain the following guarantee: we can lower bound each

[2]We let $[a]^+ := \max\{0,a\}$.

$\gamma d^2(j', S^j)$ in Equation (β_j) by $\alpha_j - \frac{\gamma}{\gamma-1} \cdot d^2(j, j')$[3], and then average all Equation (β_j) to get the guarantee.

$$\sum_j \alpha_j \leq \hat{f} + \gamma \cdot \sum_j d^2(j, i) + \frac{\gamma}{\gamma - 1} \cdot \frac{1}{|D^*|} \sum_{j<j'} d^2(j, j') \,. \tag{4}$$

Now we can exploit the specific properties of Euclidean metrics. Recall that $\frac{1}{|D^*|} \sum_{j<j'} d^2(j, j')$ is equal to the sum of the squared distances from each $j \in D^*$ to the center of mass of D^*. Hence in particular it is a valid lower bound on $\sum_{j \in D^*} d^2(j, i)$. This would imply an LMP 4-approximation by choosing $\gamma = 2$ if the same bound would hold for all the possible combinations of IC^j and DC^j (instead of the specific values that we assumed above).

Unfortunately, the latter claim is not true. In [6], no matter what the combination of IC^j and DC^j is, after lower bounding each $\gamma d^2(j', S^j)$ by Equation (2), we can always find coefficients $\{\beta_j\}_{j \in D^*}$ so that we can use $\sum_j \beta_j \cdot$(LHS of (β_j)) $\leq \sum_j \beta_j \cdot$(RHS of ($\beta_j$)) to upper bound $\sum_j \alpha_j - \hat{f} - \gamma \sum_j d^2(i, j)$ by $\frac{2\gamma}{\gamma-1} \sum_{j'<j} c_{jj'}(d^2(i, j) + d^2(i, j'))$ satisfying $\sum_{j'} c_{jj'} \leq 1$[4]. In our attempt for the Euclidean case, we can simply replace the terms of $d^2(i, j) + d^2(i, j')$ by $d^2(j, j')$ (and replace $\frac{2\gamma}{\gamma-1}$ with $\frac{\gamma}{\gamma-1}$). However, the constraint on $c_{jj'}$ is not sufficient to prove our desired Equation (4) due to the lack of symmetry in this analysis: for some combinations of IC^j and DC^j, we may have large coefficient $c_{jj'}$ for some large $d^2(j, j')$ so that the resulting upper bound exceeds $\frac{1}{|D^*|} \sum_{j<j'} d^2(j, j')$. In fact, Section 2 presents a simple counter-example showing that we cannot upper bound $\sum_{j \in D^*} \alpha_j$ by $\hat{f} + 4.1 \sum_{j \in D^*} d^2(i, j)$ with the considered algorithm.

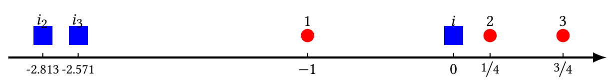

Figure 1: Counterexample for this simple attempt under $\gamma = 2$. The example is on a line. There are three clients (red circles). The facility i is the average of the clients. The facility opening cost is $\hat{f} = 22.1$. At time 8.45, client 1 becomes inactive. Let i_2 be the facility 1 directly connects to. At time ≈ 9.38, client 2 becomes inactive and indirectly connects to some facility. Then, slightly later, client 1 switches to directly connecting to i_3. This makes client 3 become inactive after time ≈ 11.02. The sum of α-values is then $\approx 28.85 = \hat{f} + 6.75 > \hat{f} + 4.15 \cdot (1^2 + (1/4)^2 + (3/4)^2)$.

Our new greedy algorithm. To handle this issue, we introduce a natural and simple, but critical update in the algorithm: every time we add a new open facility i to the current set of open facilities S, we (possibly) lower the α-values for each client j in IC to its (possibly) reduced connection cost $d^2(j, S \cup \{i\})$.

Let $\gamma = 2$. This new step immediately allows us to write a collection of new constraints for α: for any client j, and any two clients $j' \in DC^j, \ell \in IC^j$, we have

$$\alpha_\ell \leq d^2(\ell, S^j) \leq 2d^2(j', S^j) + 2d^2(j', \ell) \,. \tag{5}$$

[3]For [6], this term should be $\alpha_j - \frac{2\gamma}{\gamma-1} \cdot (d^2(i, j) + d^2(i, j'))$ using Equation (2).

[4]We assume $c_{jj'} = c_{j'j}$ for $j' > j$.

This new constraint is more powerful than Equation (3) in the sense that the latter is only a special case when $j = \ell$. With these new constraints, we can lower bound each $\gamma d^2(j', S^j)$ in Equation (β_j) by a convex combination of $\alpha_\ell - 2d^2(j', \ell)$. Then, we can show by induction on the size of D^* that we can find a convex combination for every $\gamma d^2(j', S^j)$ and a coefficient β_j for each Equation (β_j) that gives us our desired upper bound for $\sum_j \alpha_j$, Equation (4). Our choices of the convex combinations and the coefficients exploit two properties in Euclidean k-Means: for any set of clients C and another client $j \notin C$, we have $\text{cost}(C) \leq \text{cost}(C \cup \{j\})$ and $\text{cost}(C \cup \{j\}) = \frac{1}{|C|+1} \sum_{j' \in C} d^2(1, j') + \frac{|C|}{|C|+1} \text{cost}(C)$, where $\text{cost}(\cdot)$ denotes the optimal cost for a cluster (with the freedom to choose a center arbitrarily in Euclidean space). The induction steps are highly non-trivial and are presented in Lemma 13 and Subsection 3.2.

New challenges in walking between two solutions. Following the framework of [6, 13], we design a *log-adaptive* version of the above greedy algorithm; the algorithm works in $O(\log n/\epsilon^3)$ *phases* with each phase having some *flexibility* in its execution. In the log-adaptive algorithm, phase p corresponds to the time $\theta = (1+\epsilon^3)^{p-1}$, so we discretely jump θ values instead of increasing θ continuously.

The reason why we need the log-adaptive version for our bicriteria approximation algorithm (Theorem 2) is as follows. We will maintain two solutions $\mathcal{H} = (H_1, \dots, H_p)$ and $\mathcal{H}' = (H'_1, \dots, H'_{p'})$ output by the log-adaptive algorithm, where each H_i and H'_i denote the *execution history* in phase i and the total numbers of open facilities in $\mathcal{H}$ and $\mathcal{H}'$ *sandwich* k. Then for $i = 1, \dots$ inductively, we ensure that $H_i \approx H'_i$ by *walking* between two solutions, eventually leading to both solutions opening k facilities: this is made possible by the flexibility within each phase. (The base case is that the starting instance is identical except for an exponentially small difference in the facility cost f.) The walking procedure exploits $O(1)$ *free facilities* (which are open without being paid for by the dual variables) per phase, which explains the $O(\log n/\epsilon^3)$ extra centers in Theorem 2 and the need for a small number of phases.

However, our new step of lowering α-values for IC clients poses two main challenges in developing the walking procedure:

(1) **The order in which the facilities are opened matters.** The previous walking procedure of [6] crucially used the fact that, at any point in time, if two solutions opened the same set of facilities, their future executions will (almost) remain the same; in particular, the order in which the facilities were opened does not affect the future execution.
However, this property does not hold after we lower α-values for indirectly connected clients after opening each facility. Suppose an *IC* client j has an α-value of 2.1. Two facilities i_1, i_2 satisfying $d^2(j, i_1) = 1, d^2(j, i_2) = 1.1$. If we open i_1 before i_2, j will be directly connected to i_1 and its future bid to another facility i_0 will be $[2 - 2d^2(j, i_0)]^+$. On the other hand, if we open i_2 first, j will be indirectly connected to i_1 because we first lower α_j to $d^2(j, i_2) = 1.1$. The future bid in this execution is then $[1.1 - 2d^2(j, i_0)]^+$. This large difference in future bids could make a significant difference in future executions.

(2) **Discontinuity in the bids.** Another crucial invariant of the previous walking procedure [6] was that, the same client

j's bid towards any facility i is almost the same between the two solutions $\mathcal{H} = (H_1, \dots, H_p)$ and $\mathcal{H}' = (H'_1, \dots, H'_{p'})$ we maintain. This *continuity* ensures that even if we *mix and match* different parts of the two solutions (e.g., $\mathcal{H}'' = (H_1, \dots, H_\ell, H'_{\ell+1}, \dots, H'_{p'})$ for some ℓ), the resulting solution, which was never an outcome of the log-adaptive algorithm, can be considered as (almost) a valid execution of the log-adaptive algorithm.
However, this property does not hold in our new algorithm. Suppose $\alpha_j = 1 - 2^{-n}$ and $\alpha'_j = 1 + 2^{-n}$ at some time point of the algorithm and there is a newly opened facility i such that $d^2(j,i) = 0.5$. Then j will be indirectly connected to i in the first execution, while j will be directly connected in the second case. As discussed above, the bid of j in the future can be completely different.

Log-adaptive Algorithm. To resolve the first challenge, we will use a sophisticated log-adaptive version of our algorithm. Instead of immediately lowering α-value for IC clients after a facility is opened, we only lower α-values at the end of a phase. This ensures that the order in which facilities are opened in the same phase no longer matters. This turns out to be sufficient for the walking part.

Fractional connections. To resolve the second challenge, we will introduce fractional connections to the clients. To be more specific, we do not change the facility each client connects to. Instead, we allow each client to be split into several copies, in which some copies are directly connected and at most one copy is indirectly connected to the facility. Then, the bid of a client j to a facility i is the average among all its copies.

This concept of fractional connections allows us to modify the fractions of each client in each copy in order to make its bid "continuous" in two different executions. To ensure that this modification does not hurt the guarantees of the log-adaptive algorithm, we only split some directly connected copies with $\alpha \le 2d^2(j,S) + O(2^{-n})$, which turns out to be sufficient to ensure the continuity in the bids. See the full version for the definition of the split and the analysis.

Organizations for our paper. In Section 3, we will present the greedy LMP 4-approximation for Facility Location with squared metric distances. In Section 5 of the full version, we will prove Theorem 2 by presenting the log-adaptive algorithm and the algorithm that walks between two solutions. In Section 6 of the full version, we will prove Theorem 3 by presenting the $4+O(\sqrt{\epsilon})$-approximation for stable instances.

3 Greedy Algorithm

In this section, we present the new algorithm for Facility Location with Euclidean squared distances. This algorithm is not directly used in our $(4+\epsilon)$-approximation for Euclidean k-Means, but it allows us to present the main ideas behind the log-adaptive algorithm presented in the full version in a simpler setting. Given an instance $(D, \mathcal{F}, d, f)$ for Facility Location with squared Euclidean distances, we let $\hat{f} = 4f$. The following Algorithm 1 is our new greedy algorithm achieving an LMP 4-approximation. As standard in the area, we present the algorithm in a continuous version where there is a variable θ (interpreted as *time*) that grows continuously from 0 to $+\infty$, and certain quantities grow at the same speed as θ for a certain interval of time. It is easy to discretize this algorithm.

We say that the clients in A are active, that the clients in IC are indirectly connected, and that the clients in DC are directly connected.

Observation 4. *At any time of the algorithm, for any $j \in IC$, we have $\alpha_j = d^2(j,S)$.*

Because of the fact that we freeze the α values when a client becomes directly connected, we can show that the α values are non-increasing after connection. For simplicity, for some intermediate time θ, we will use $\alpha_j(\theta)$ to denote the α value of client $j \in D$ at that time. Further, for each $j \in D$, we define $\hat{\alpha}_j$ to be its α value at the first time when j is connected to some facility.

Lemma 5. *For any client j, the value of $\alpha_j(\theta)$ is non-increasing in $\theta \in [\hat{\alpha}_j, +\infty)$.*

Proof. This follows the fact that we only modify α_j after time $\hat{\alpha}_j$ when some facility i is opened and $\alpha_j > d^2(j,i)$. Then the resulting value of α_j is lowered to $d^2(j,i)$. □

Let $(\alpha^*_j)_{j \in D}$ be the final vector of α-values.

Corollary 6. *For each client j, we have $\alpha^*_j \le \hat{\alpha}_j$.*

Moreover, for each client j that is finally directly connected, α^*_j can be written as the sum of $2d^2(j,S)$ and the bids of j to the open facilities. Therefore, we can obtain an upper bound for our algorithm's total cost using the α^*-values.

Lemma 7. *At the end of the algorithm, $\sum_{j\in D} d^2(j,S) + |S| \cdot \hat{f} \le \sum_{j \in D} \alpha^*_j$.*

Proof. First, observe that it is sufficient to prove that at the end of the execution, we have

$$\sum_{j \in DC} \alpha_j \ge \sum_{j \in DC} 2d^2(j,S) + \sum_{i \in S} \hat{f}. \tag{6}$$

Indeed, at the end of the execution, $A = \emptyset$, and for every $j \in IC$, $\alpha^*_j = d^2(j,S)$. To prove this, we prove that at any point in the execution, (6) holds.

The equality is initially true since $DC = \emptyset$ and $S = \emptyset$. Furthermore, since whenever the α-value of a client j reaches $d^2(j,S)$ it stops growing, no client is added to DC excluding when a new facility is opened (i.e., added to S).

We now consider what happens when we open a facility i, i.e., add it to S. Let (α, S, A, IC, DC) be the state right before opening i, and set $DC' = \{j \in A \cup IC : \alpha_j \ge 2d^2(i,j)\}$, and $X = \{j \in DC : d^2(j,i) < d^2(j,S)\}$. The change of cost of the right-hand side of (6) is at most

$$\hat{f} + \sum_{j \in DC'} 2d^2(i,j) + \sum_{j \in X} 2(d^2(i,j) - d^2(j,S)).$$

Since the algorithm decided to open i, we also have

$$\hat{f} \le \sum_{j \in DC'} (\alpha_j - 2d^2(i,j)) + \sum_{j \in X} (2d^2(j,S) - 2d^2(i,j)).$$

We thus get that the change of cost of the right-hand side is at most $\sum_{j \in DC'} \alpha_j$, which is the change of the left-hand side of (6). □

Algorithm 1 (Greedy Algorithm).

Initialization: $A = D$, $IC = DC = S = \emptyset$, and $\alpha_j = 0$ for every $j \in D$.

While $A \neq \emptyset$, uniformly grow the α_j of every client in A until:

(1) For some unopened facility $i \in \mathcal{F} \setminus S$, we have

$$\sum_{j \in A \cup IC} \left[\alpha_j - 2d^2(j,i)\right]^+ + \sum_{j \in DC} \left[2d^2(j,S) - 2d^2(j,i)\right]^+ = \hat{f}.$$

In this case, we open facility i. For each client $j \in DC$, if $d^2(j,S) > d^2(j,i)$, we reconnect j to i. For each client $j \in A \cup IC$,

- If $\alpha_j \geq 2d^2(j,i)$, we (re)connect j to i and move j to DC.
- Else, if $\alpha_j \geq d^2(j,i)$, we (re)connect j to i, set α_j to be $d^2(j,i)$, and move j to IC (if $j \in A$).

(2) For some client $j \in A$, we have

$$\alpha_j = d^2(j,S).$$

In this case, we move j to IC and connect j to the closest facility to it in S.

The last ingredient for the analysis is to show that α^* can be scaled to a dual feasible solution. The proof of the following lemma is given in Section 3.1.

Lemma 8. *For any facility $i \in \mathcal{F}$, we have $\sum_{j \in D}[\frac{1}{4} \cdot \alpha_j^* - d^2(j,i)]^+ \leq f$.*

Now, we use Lemmas 7 and 8 to establish the LMP approximation ratio.

Theorem 9. *Algorithm 1 is an LMP 4-approximation for Facility Location with squared Euclidean distances.*

Proof. By Lemma 8, $\alpha^*/4$ is a feasible solution to the dual of the standard Facility Location LP, hence $\sum_{j \in D} \alpha_j^* \leq 4 \cdot \mathrm{opt}_{LP}(f)$ by weak duality. Our solution S satisfies:

$$\sum_{j \in D} d^2(j,S) + 4|S|f \overset{\text{Lem. 7}}{\leq} \sum_{j \in D} \alpha_j^* \leq 4 \cdot \mathrm{opt}_{LP}(f). \qquad \square$$

3.1 Dual Feasibility

In this subsection, we prove Lemma 8. We will use $[n]$ to denote the set $\{1, 2, \dots, n\}$ and $[l : r]$ to denote the set $\{l, l+1, \dots, r\}$. From now on, we fix any facility $i \in \mathcal{F}$. Suppose that there are s clients such that $\alpha_j^* \geq 4 \cdot d^2(j,i)$, and we index them by $[s]$ and order them in the increasing order of the time they become inactive. That is, we have $\hat{\alpha}_1 \leq \hat{\alpha}_2 \leq \cdots \leq \hat{\alpha}_s$. Then, our goal (which will be formally shown in Corollary 14) becomes proving

$$\sum_{j \in [s]} \alpha_j^* \leq 4f + \sum_{j \in [s]} 4d^2(j,i) = \hat{f} + 4 \cdot \sum_{j \in [s]} d^2(j,i)\,. \tag{7}$$

Consider the time $\hat{\alpha}_t - \epsilon$ for each client t, where $\epsilon > 0$ is sufficiently small so that neither condition in the algorithm is satisfied between time $\hat{\alpha}_t - \epsilon$ (inclusive) and $\hat{\alpha}_t$ (exclusive). For simplicity, we define DC^t as the set of directed connected clients in $[t]$ (i.e., $DC \cap [t]$) at time $\hat{\alpha}_t - \epsilon$, and IC^t as the set of indirectly connected or active clients in $[t]$ (i.e., $(A \cup IC) \cap [t]$) at time $\hat{\alpha}_t - \epsilon$. Moreover, we use S^t to denote the set S at time $\hat{\alpha}_t - \epsilon$. Consider any client $j' \in IC^t$. As $j' \leq t$, we have $\hat{\alpha}_{j'} \leq \hat{\alpha}_t$ by our assumption. Next, we prove that $\alpha_{j'}^* \leq d^2(j', S^t)$ by discussing two cases:

- If $\hat{\alpha}_{j'} < \hat{\alpha}_t$ (since ϵ is sufficiently small, we have $\hat{\alpha}_{j'} \leq \hat{\alpha}_t - \epsilon$), then j' is indirectly connected at time $\hat{\alpha}_t - \epsilon$. Because of Observation 4 and Lemma 5 we have $\alpha_{j'}^* \leq \alpha_{j'}(\hat{\alpha}_t - \epsilon) = d^2(j', S^t)$.
- Otherwise (i.e., if $\hat{\alpha}_{j'} = \hat{\alpha}_t$), we can prove $\hat{\alpha}_t - \epsilon < d^2(j', S^t)$ by contradiction. Suppose that $\hat{\alpha}_t - \epsilon \geq d^2(j', S^t)$. Then, j' will be connected to some opened facility using the second condition in Algorithm 1 at time $\hat{\alpha}_t - \epsilon$, contradicting that j' is active at time $\hat{\alpha}_t - \epsilon$. Since ϵ can be arbitrarily small here, we have $\hat{\alpha}_{j'} = \hat{\alpha}_t \leq d^2(j', S^t)$. Because of Corollary 6, we get $\alpha_{j'}^* \leq d^2(j', S^t)$.

Note that $d(j', S^t) \leq d(j, S^t) + d(j, j')$ and $(x+y)^2 \leq 2(x^2+y^2)$ for any $x, y \in \mathbb{R}_{\geq 0}$. For each $t \in [s]$, $j \in DC^t$, $j' \in IC^t$, we have the following constraint:

$$\alpha_{j'}^* \leq 2 \cdot d^2(j, S^t) + 2 \cdot d^2(j, j')\,. \tag{$\phi_{j,j'}^{(t)}$}$$

Consider the time $\hat{\alpha}_t - \epsilon$ for each $t \in [s]$. Note that we assume that $\alpha_t^* \geq 4 \cdot d^2(t,i)$. Because of Corollary 6 and because ϵ is sufficiently small, we have $\hat{\alpha}_t - \epsilon \geq \alpha_t^* - \epsilon > d^2(t,i)$. If facility i is opened at time $\hat{\alpha}_t - \epsilon$, t should be connected to some facility at time $\hat{\alpha}_t - \epsilon$, contradicting our assumption that $\hat{\alpha}_t$ denotes the time client t is connected for the first time. Therefore, the total bid to i at this time stamp should be strictly less than $\hat{f}$, i.e.,

$$\sum_{j \in DC^t} \left[2d^2(j, S^t) - 2d^2(j,i)\right]^+ + \sum_{j \notin DC^t} \left[\alpha_j(\hat{\alpha}_t - \epsilon) - 2d^2(j,i)\right]^+ < \hat{f}.$$

If $j \geq t$, we have $\alpha_j(\hat{\alpha}_t - \epsilon) = \hat{\alpha}_t - \epsilon \geq \alpha_t^* - \epsilon$. Otherwise, if $j < t$ and $j \notin DC^t$, we have either j being indirectly connected (i.e., $\hat{\alpha}_j \leq \hat{\alpha}_t - \epsilon$), or $\alpha_j(\hat{\alpha}_t - \epsilon) = \hat{\alpha}_t - \epsilon$. Because of Lemma 5 (for the first case) and Corollary 6, $\hat{\alpha}_j \leq \hat{\alpha}_t$ (for the second case), we have $\alpha_j^* - \epsilon \leq \alpha_j(\hat{\alpha}_t - \epsilon)$. Recall that $IC^t \cup DC^t = [t]$ according to our definition for IC^t, DC^t. Because ϵ is sufficiently small, we have

$$\sum_{j \in DC^t} \left[2 \cdot d^2(j, S^t) - 2 \cdot d^2(j,i)\right]^+ + \sum_{j \in IC^t} \left[\alpha_j^* - 2 \cdot d^2(j,i)\right]^+ + \sum_{j > t} [\alpha_t^* - 2 \cdot d^2(j,i)]^+ \leq \hat{f}.$$

Since $[x]^+ \geq x$, for each $t \in [s]$, we have the following constraint:

$$(s - t + 1)\alpha_t^* + \sum_{j \in IC^t \setminus \{t\}} \alpha_j^* + \sum_{j \in DC^t} 2d^2(j, S^t) \leq \hat{f} + 2 \sum_{j \in [s]} d^2(j,i)\,. \tag{β_t}$$

Next, we combine the two constraints ($\phi^{(t)}_{j,j'}$) and (β_t) to obtain the dual feasibility. We will rewrite $2d^2(j,S^t)$ in (β_t) by a convex combination of $\alpha^*_{j'} - 2d^2(j,j')$ for $j' \in IC^t$ such that $j' > j$. We will use the notation $\Phi^{(t)}_j$ and $\phi^{(t)}_{j,j'}$ as follows:

$$\Phi^{(t)}_j = \sum_{j' \in IC^t : j' > j} \phi^{(t)}_{j,j'} \cdot (\alpha^*_{j'} - 2d^2(j,j')) . \tag{8}$$

Then, using ($\phi^{(t)}_{j,j'}$), we have the following lemma.

Lemma 10. *Suppose $\sum_{j' \in IC^t : j' > j} \phi^{(t)}_{j,j'} = 1$ for any $t \in [s], j \in DC^t$. Then, we have $2 \cdot d^2(j,S^t) \geq \Phi^{(t)}_j$.*

Proof. We have

$$\begin{aligned} 2 \cdot d^2(j,S^t) &= \sum_{j' \in IC^t : j' > j} \phi^{(t)}_{j,j'} \cdot 2d^2(j,S^t) \\ &\geq \sum_{j' \in IC^t : j' > j} \phi^{(t)}_{j,j'} \cdot (\alpha^*_{j'} - 2d^2(j,j')) \\ &\qquad\qquad \text{(By Equation } (\phi^{(t)}_{j,j'})) \\ &= \Phi^{(t)}_j. \end{aligned}$$

□

For simplicity, we will define $\phi^{(t)}_{j,j'} = 0$ for $j' \notin [j+1 : s] \cap IC^t$. With this notation, we will prove that the two constraints, ($\phi^{(t)}_{j,j'}$) and (β_t) (replacing $d^2(j,S^t)$ by $\Phi^{(t)}_j$ using Lemma 10), together imply an upper bound on the sum of α^*_t. Technically, we will show the following Lemma 13, which is independent of the context of our algorithm and only uses the properties of Euclidean metrics. Because this lemma will also be used in future sections of the full version, we prove a more general weighted version here. More specifically, in Lemma 13, we will consider the weighted continuous Euclidean k-Means problem (Definition 11), and we will use a characterization of the optimal cost of a cluster consisting of a set of clients among all possible choices of the center (Lemma 12). We defer the proof of Lemma 12 to the full version. (In the context of this section, readers can assume that $w(j) = 1$ for every $j \in [s]$, and omit Bullets D and E.)

Definition 11 (weighted continuous Euclidean k-Means)**.** *Let $\dim > 0$ be the number of dimensions of the Euclidean space. The input consists of a set of clients D, locations of the clients $x : D \to \mathbb{R}^{\dim}$ and weights $w : D \to \mathbb{R}_{\geq 0}$. The goal of the weighted continuous Euclidean k-Means problem is to find a set of k centers $S \subseteq \mathbb{R}^{\dim}$ such that $|S| = k$ and the following objective is minimized:*

$$\sum_{j \in D} w(j) \cdot \min_{c \in S} \|x_j - c\|_2^2 .$$

In particular, an instance is unweighted if $w(j) = 1$ for any client $j \in D$.

Lemma 12 (single-cluster cost of weighted continuous Euclidean k-Means)**.** *Suppose the set $C \subseteq D$ forms a cluster. The optimal center is the weighted average of the points $\frac{\sum_{j \in C} w(j) \cdot x_j}{\sum_{j \in C} w(j)}$, and the cost is*

$$\text{cost}_w(C) = \frac{\sum_{j,j' \in \binom{C}{2}} w(j) \cdot w(j') \cdot \|x_j - x_{j'}\|_2^2}{\sum_{j \in C} w(j)} .$$

In particular, if the instance is unweighted, we will use the notation $cost(C)$ for simplicity.

Lemma 13. *Fix any integer $s \geq 1$ and any Euclidean metric d defined on $[s]$. For each $t \in [s]$, let $w(t) > 0$ be any positive weight. For each $t \in [s]$, fix any partition of $[t]$ by $DC^t \sqcup IC^t = [t]$ such that*

- *For each $t \in [s]$, $t \in IC^t$;*
- *For each $j, t, t' \in [s]$ satisfying $j < t < t'$, if $j \in DC^t$, then $j \in DC^{t'}$.*

Then, there exist non-negative real numbers $\{\phi^{(t)}_{j,j'}\}_{t \in [s], j \in DC^t, j' \in IC^t}$ and $\{\beta_j\}_{j \in [s]}$ such that

A. *For each $t \in [s], j \in DC^t$, the corresponding ϕ variables sum to 1:*

$$\sum_{j' \in IC^t : j' > j} \phi^{(t)}_{j,j'} = 1 , \tag{9}$$

and $\phi^{(t)}_{j,j'} = 0$ for any $j' \in IC^t$ such that $j' < j$.

B. *All β variables sum to 1:*

$$\sum_{i \in [s]} \beta_i = 1 . \tag{10}$$

C. *Let $\{\alpha^*_j\}_{j \in [s]}$ be any vector with non-negative entries. Let*

$$\Phi^{(t)}_j = \sum_{j' \in IC^t : j' > j} \phi^{(t)}_{j,j'} \cdot (\alpha^*_{j'} - c \cdot d^2(j,j'))$$

for some constant $c > 1$. Then, the following inequality holds:

$$\begin{aligned} \sum_{t \in [s]} \beta_t \cdot \left(\sum_{j \in IC^t \setminus \{t\}} w(j)\alpha^*_j + \sum_{j \in DC^t} w(j)\Phi^{(t)}_j + \sum_{j \geq t} w(j)\alpha^*_t \right) \\ \geq \sum_{t \in [s]} w(t)\alpha^*_t - c \cdot cost_w([s]). \end{aligned} \tag{11}$$

In particular, we can ensure the following properties:

D. *For any $j \in IC^s$, we have $\beta_j = 0$.*

E. *For any $j \in DC^s$, we have $\phi^{(s)}_{j,j'}/w(j') \leq \phi^{(s)}_{j,j''}/w(j'')$ for any $j' < j'' \in IC^s$.*

The upper bound of the sum of α^*_t is then a direct corollary for unit weights, which implies our goal of this subsection.

Corollary 14. *For any $S \subseteq D$ and $i \in \mathcal{F}$, we have*

$$\sum_{j \in [s]} \alpha^*_j \leq \hat{f} + 2 \cdot \sum_{j \in [s]} d^2(i,j) + 2 \cdot cost([s]) , \tag{12}$$

where $cost([s]) := \frac{1}{s} \sum_{x,y \in \binom{[s]}{2}} \|x - y\|_2^2$.

In particular, the above corollary implies Equation (7).

Proof. Suppose that $\{\beta_j\}_{j \in [s]}, \{\phi^{(t)}_{j,j'}\}_{t \in [s], j \in DC^t, j' \in IC^t}$ denote the parameters obtained in Lemma 13. Using Lemma 10 and Bullets A and C of Lemma 13, we have

$$\sum_{t \in [s]} \beta_t \cdot (\text{LHS of Equation } (\beta_t)) \geq \sum_{t \in [s]} \alpha^*_t - 2\text{cost}([s]) .$$

According to Bullet B of Lemma 13, we have

$$\sum_{t \in [s]} \beta_t \cdot (\text{LHS of Equation } (\beta_t)) \leq \hat{f} + 2 \sum_{j \in [s]} d^2(j,i).$$

Therefore,

$$\sum_{t\in[s]} \alpha_t^* - 2\cdot \mathrm{cost}([s]) \le \hat{f} + 2\cdot \sum_{j\in[s]} d^2(j,i)\,,$$

obtaining Equation (12). Further, because cost$([s])$ denotes the optimal cost of opening a single center for clients in $[s]$ (with the freedom of choosing an arbitrary center in the Euclidean space), we have $\mathrm{cost}([s]) \le \sum_{j\in[s]} d^2(j,i)$, and thus $\sum_{t\in[s]} \alpha_t^* \le \hat{f} + 4\cdot \sum_{j\in[s]} d^2(j,i)$, obtaining Equation (7). □

Finally, in Section 3.2, we prove Lemma 13.

3.2 Proof of Lemma 13

A stronger argument. We will show a stronger version for Bullet C of Lemma 13. We will use B_t to denote the t-th term on the LHS of Equation (11):

$$B_t = \beta_t \cdot \left(\sum_{j\in IC^t\setminus\{t\}} w(j)\cdot \alpha_j^* + \sum_{j\in DC^t} w(j)\cdot \Phi_j^{(t)} + \sum_{j\ge t} w(j)\cdot \alpha_t^* \right).$$

Suppose $\eta_t(j')$ denotes the coefficient of $\alpha_{j'}^*$ in B_t, i.e.,

$$\eta_t(j') = \begin{cases} \beta_t\cdot\left(\sum_{\ell\ge t} w(\ell) + \sum_{j\in DC^t} w(j)\cdot \phi_{j,j'}^{(t)}\right) & \text{if } j'=t;\\ \beta_t\cdot\left(w(j') + \sum_{j\in DC^t} w(j)\cdot\phi_{j,j'}^{(t)}\right) & \text{if } j'\in IC^t\setminus\{t\};\\ 0 & \text{otherwise.}\end{cases} \tag{13}$$

For any $s'\in[s]$ and $j'\in[s]$, we use $w_{s'}(j')$ to denote the total coefficient of $\alpha_{j'}^*$ in the first s' terms:

$$w_{s'}(j') = \sum_{t\in[s']} \eta_t(j')\,. \tag{14}$$

Further, The stronger version for Bullet C is two-part: (i) for any $j\in[s]$,

$$w_s(j) = w(j)\,; \tag{C(i)}$$

(ii) for any prefix of size $s'\in[s]$,

$$\sum_{t\in[s']} B_t \ge \left(\sum_{j\in[s]} w_{s'}(j)\cdot\alpha_j^*\right) - \frac{c\sum_{j\in[s]} w_{s'}(j)}{\sum_{j\in[s]} w(j)}\cdot \mathrm{cost}_{w_{s'}}([s])\,. \tag{C(ii)}$$

For simplicity, we will call two parts C(i) and C(ii) respectively. If we take $s'=s$ in C(ii) and use the guarantee in C(i), we get $\sum_{t\in[s]} B_t \ge \sum_{j\in[s]} w(j)\cdot\alpha_j^* - c\cdot \mathrm{cost}_w([s])$, which is exactly the same inequality as the original Bullet C of Lemma 13. Next, we will prove this stronger argument by induction.

Base case. The base case is when $s=1$, where we can pick $\beta_1=1$ (and since $DC^1=\emptyset$, there is no parameter $\phi_{j,j'}^{(t)}$). All the five bullets are then straightforward.

Induction step. Consider any $s_0\ge 2$. Suppose that we have shown the stronger argument for all $s<s_0$. Consider $s=s_0$ next. For the set $[2:s]$, it is clear that d is still a Euclidean metric on $[2:s]$ and the sets $(DC^t\setminus\{1\})\sqcup(IC^t\setminus\{1\}) = [2:s]$ form a partition of $[2:s]$ for any $t\in[2:s]$. Moreover, $DC^t\setminus\{1\}$ and $IC^t\setminus\{1\}$ still satisfy the conditions of Lemma 13. Suppose W and $\tilde{W}$ denote the total weights after and before we consider the first element.

$$W = \sum_{j\in[s]} w(j)\,,\qquad\text{and}\qquad \tilde{W} = \sum_{j\in[2:s]} w(j)\,.$$

According to the induction hypothesis, we can find non-negative parameters $\{\tilde\beta_j\}_{j\in[2:s]}$ and $\{\tilde\phi_{j,j'}^{(t)}\}_{t\in[2:s],j\in DC^t\setminus\{1\},j'\in IC^t\setminus\{1\}}$ restricted to the set $[2:s]$, and $\tilde\eta_t(j), \tilde{w}_{s'}(j)$ defined accordingly (also, restricted to $[2:s]$) such that all the following hold.

- For any $t\in[2:s]$ and $j\in DC^t\setminus\{1\}$, $\tilde\Phi_j^{(t)}$ follows the definition:
$$\tilde\Phi_j^{(t)} = \sum_{j'\in IC^t\setminus\{1\}} \tilde\phi_{j,j'}^{(t)}\cdot(\alpha_{j'}^* - 2\cdot d^2(j,j')).$$
- For each $t\in[2:s]$, $j\in DC^t\setminus\{1\}$, we have
$$\sum_{j'\in IC^t: j'>j} \tilde\phi_{j,j'}^{(t)} = 1.$$
Further, if $j'<j$ and $j'\in IC^t\setminus\{1\}$, we have $\tilde\phi_{j,j'}^{(t)}=0$.
- $\sum_{t\in[2:s]}\tilde\beta_t = 1$.
- The t-th term $\tilde B_t$ (given any non-negative vector $\{\alpha_j^*\}_{j\in[2:s]}$) is defined by
$$\tilde B_t = \tilde\beta_t\cdot\left(\sum_{j\in IC^t\setminus\{1\}} w(j)\alpha_j^* + \sum_{j\in DC^t\setminus\{1\}} w(j)\tilde\Phi_j^{(t)} + \sum_{j\ge t} w(j)\alpha_t^*\right). \tag{15}$$
- The definitions of $\tilde\eta,\tilde w$ follow:
$$\forall t,j'\in[2:s], \tilde\eta_t(j') = \begin{cases} \tilde\beta_t\cdot\left(\sum_{\ell\ge t} w(\ell) + \sum_{j\in DC^t\setminus\{1\}} w(j)\tilde\phi_{j,j'}^{(t)}\right) & \text{if } j'=t\\ \tilde\beta_t\cdot\left(w(j') + \sum_{j\in DC^t\setminus\{1\}} w(j)\tilde\phi_{j,j'}^{(t)}\right) & \text{if } j'\in IC^t\setminus\{1,t\}\\ 0 & \text{otherwise}\end{cases} \tag{16}$$
$$\forall s',j'\in[2:s], \tilde w_{s'}(j') = \sum_{t\in[2:s']} \tilde\eta_t(j')\,. \tag{17}$$
Then, Equation (C(i)) gives $\tilde w_s(j') = w(j')$ for any $j'\in[2:s]$.
- For any non-negative vector $\{\alpha_j^*\}_{j\in[2:s]}$ and any $s'\in[2:s]$, Equation (C(ii)) can be rewritten as
$$\sum_{t\in[2:s']}\tilde B_t \ge \sum_{j\in[2:s]} \tilde w_{s'}(j)\cdot\alpha_j^* - \frac{2\sum_{j\in[2:s]}\tilde w_{s'}(j)}{\tilde W}\cdot \mathrm{cost}_{\tilde w_{s'}}([2:s])\,.$$

Next, we will construct new parameters $\{\phi_{j,j'}^{(t)}\}_{t\in[s],j\in DC^t,j'\in IC^t}$ and $\{\beta_j\}_{j\in[s]}$ that satisfy our induction hypothesis. For any $t\in[2:s]$, $j\in DC^t\setminus\{1\}$, $j'\in IC^t\setminus\{1\}$, we let $\phi_{j,j'}^{(t)} = \tilde\phi_{j,j'}^{(t)}$. Now, only the following ϕ variables remained undefined: $\phi_{1,j'}^{(t)}$ for $t\in[2:s]$ and $j'\in IC^t$ such that $1\in DC^t$. Then, for each $t\in[2:s]$, our definitions of β_t and these ϕ parameters follow two cases:

- If $1\in IC^t$, we let $\beta_t=\tilde\beta_t$ and let $\phi_{j,1}^{(t)}=0$ for any $j\in DC^t$.
- Otherwise (if $1\in DC^t$), we let $\beta_t = \frac{\tilde W}{W}\cdot\tilde\beta_t$ and $\phi_{1,j'}^{(t)} = \frac{1}{\tilde W}\cdot\frac{\tilde\eta_t(j')}{\tilde\beta_t}$ for each $j'\in IC^t$.

Finally, we let $\beta_1 = 1 - \sum_{t\geq 2}\beta_t$.

This definition immediately gives us (i) $\beta_t \geq 0$ for each $t \in [2:s]$, (ii) $\phi^{(t)}_{j,j'} \geq 0$ for each $t \in [s], j \in DC^t, j' \in IC^t$, and (iii) Bullet B of Lemma 13. Further, according to the definition and our induction hypothesis, we have $\sum_{j\in[2:s]}\beta_j \leq \sum_{j\in[2:s]}\tilde{\beta}_j = 1$. Therefore, we have $\beta_1 \geq 0$. In addition, if $1 \in IC^s$, we have $\beta_1 = 0$ because $1 \in IC^t$ for any $t \geq 2$ and $\sum_{j\in[2:s]}\beta_j = \sum_{j\in[2:s]}\tilde{\beta}_j = 1$. This, together with our definition that $\beta_t = \tilde{\beta}_t$ or $\beta_t = \frac{\tilde{W}}{W}\cdot\tilde{\beta}_t$, shows Bullet D of Lemma 13.

Next, we will show that Bullets A, E in Lemma 13 and (C(i)) and (C(ii)) in the stronger argument hold.

Bullet A of Lemma 13. For $t \in [2:s]$ and $j \in DC^t$ such that $j \neq 1$, Bullet A follows the fact that $\phi^{(t)}_{j,j'} = \tilde{\phi}^{(t)}_{j,j'}$ for any $j' \in IC^t$ and our definition that $\phi^{(t)}_{j,1} = 0$ if $1 \in IC^t$. For each $t \in [2:s]$ such that $1 \in DC^t$, according to our definition of $\phi^{(t)}_{1,j'}$, we have

$$
\begin{aligned}
\sum_{j'\in IC^t}\phi^{(t)}_{1,j'} &= \sum_{j'\in IC^t}\frac{1}{\tilde{W}}\cdot\frac{\tilde{\eta}_t(j')}{\tilde{\beta}_t}\\
&= \frac{\sum_{\ell\geq t}w(\ell) + \sum_{j\in DC^t\setminus\{1\}}w(j)\cdot\tilde{\phi}^{(t)}_{j,t}}{\tilde{W}}\\
&\quad + \sum_{j'\in IC^t\setminus\{t\}}\frac{w(j') + \sum_{j\in DC^t\setminus\{1\}}w(j)\cdot\tilde{\phi}^{(t)}_{j,j'}}{\tilde{W}} && \text{(Equation (16))}\\
&= \frac{\sum_{j\in IC^t\cup[t+1:s]}w(j)}{\tilde{W}} + \sum_{j\in DC^t\setminus\{1\}}w(j)\cdot\sum_{j'\in IC^t}\frac{\tilde{\phi}^{(t)}_{j,j'}}{\tilde{W}}\\
&= \frac{\sum_{j\in IC^t\cup[t+1:s]}w(j)}{\tilde{W}} + \sum_{j\in DC^t\setminus\{1\}}\frac{w(j)}{\tilde{W}} && \text{(induction hypothesis)}\\
&= \frac{\sum_{j\in[2:s]}w(j)}{\tilde{W}} = 1\,.
\end{aligned}
$$

Stronger argument (C(i)). Next, we assume that $\Phi^{(t)}_j, B_t, \eta_t, w_{s'}$ are defined through Equations (13) and (14). First, we prove that $\eta_t(j') = \tilde{\eta}_t(j')$ for any $t, j' \in [2:s]$ by discussing two cases.

- If $1 \in IC^t$, $\beta_t = \tilde{\beta}_t$ and we don't have new parameters $\phi^{(t)}_{1,j'}$. Therefore, $\eta_t(j') = \tilde{\eta}_t(j')$ by its definition Equation (13).
- If $1 \in DC^t$, $\beta_t = \frac{\tilde{W}}{W}\cdot\tilde{\beta}_t$. Therefore, according to Equation (13), we have $\eta_t(j') = \frac{\tilde{\beta}_t}{\beta_t}\cdot\tilde{\eta}_t(j') + \beta_t\cdot w(1)\cdot\phi^{(t)}_{1,j'} = \frac{W}{\tilde{W}}\cdot\tilde{\eta}_t(j') + \frac{\tilde{W}}{W}\cdot\tilde{\beta}_t\cdot w(1)\cdot\frac{\tilde{\eta}_t(j')}{\tilde{W}\tilde{\beta}_t} = \frac{\tilde{W}+w(1)}{W}\cdot\tilde{\eta}_t(j') = \tilde{\eta}_t(j')$.

Note that $\eta_1(j') = 0$ for any $j' \geq 2$ because $j' \notin IC^1 \cup DC^1$. This implies $w_{s'}(j') = \tilde{w}_{s'}(j')$ for any $j', s' \in [2:s]$ and thus $w_s(j') = w(j')$ for any $j' \in [2:s]$. Further, we prove $w_s(1) = w(1)$ as follows.

$$
\begin{aligned}
w_s(1) &= \beta_1\cdot W + w(1)\cdot\sum_{t\geq 2:1\in IC^t}\beta_t\\
&= W\cdot\left(1 - \sum_{t\geq 2}\beta_t\right) + w(1)\sum_{t\geq 2:1\in IC^t}\beta_t\\
&= W\cdot\left(1 - \sum_{t\geq 2}\tilde{\beta}_t + \sum_{t\geq 2:1\in DC^t}\frac{w(1)}{W}\cdot\tilde{\beta}_t\right) + w(1)\sum_{t\geq 2:1\in IC^t}\tilde{\beta}_t\\
&= w(1)\cdot\sum_{t\geq 2:1\in DC^t}\tilde{\beta}_t + w(1)\sum_{t\geq 2:1\in IC^t}\tilde{\beta}_t = w(1)\,.
\end{aligned}
$$
$$\text{(induction hypothesis: } \textstyle\sum_{t\in[2:s]}\tilde{\beta}_t = 1)$$

Stronger argument (C(ii)). Next, we fix any non-negative vector $\{\alpha^*_j\}_{j\in[s]}$. Note that there exists some threhold $t_0 \geq 2$ such that $1 \in IC^t$ for any $t \in [2:t_0-1]$ and $1 \in DC^t$ for any $t \in [t_0:s]$. Recall the definition of B_t:

$$
B_t = \beta_t\cdot\left(\sum_{j\in IC^t\setminus\{t\}}w(j)\cdot\alpha^*_j + \sum_{j\in DC^t}w(j)\cdot\Phi^{(t)}_j + \sum_{j\geq t}w(j)\cdot\alpha^*_t\right).
$$

Next, we prove Equation (C(ii)) by considering two cases: whether $s' < t_0$ or not.

The simpler case here is that $s' < t_0$. In this case, $\beta_t = \tilde{\beta}_t$ for any $t \in [2:s']$ and the Φ terms in the first s' terms do not change at all, i.e., $\Phi^{(t)}_j = \tilde{\Phi}^{(t)}_j$ for any $t \in [2:s']$ and $j \in DC^t$. Therefore, $B_t = \tilde{B}_t + \tilde{\beta}_t\cdot w(1)\cdot\alpha^*_1$ for any $t \in [2:s']$. According to our induction hypothesis, we have

$$
\begin{aligned}
\sum_{t\in[s']}B_t &= B_1 + \sum_{t\in[2:s']}(\tilde{B}_t + \beta_t\cdot w(1)\cdot\alpha^*_1)\\
&= \left(\beta_1\cdot W + \sum_{t\in[2:s']}w(1)\cdot\beta_t\right)\cdot\alpha^*_1 + \sum_{t\in[2:s']}\tilde{B}_t\\
&\geq w_{s'}(1)\alpha^*_1 + \sum_{j\in[2:s]}\tilde{w}_{s'}(j)\alpha^*_j\\
&\qquad - \frac{2\sum_{j\in[2:s]}\tilde{w}_{s'}(j)}{\tilde{W}}\cdot\mathrm{cost}_{\tilde{w}_{s'}}([2:s])\\
&= \sum_{j\in[s]}w_{s'}(j)\cdot\alpha^*_j - \frac{2\sum_{j\in[2:s]}\tilde{w}_{s'}(j)}{\tilde{W}}\cdot\mathrm{cost}_{\tilde{w}_{s'}}([2:s])
\end{aligned}
$$

Note that $w_{s'}(1) \geq w(1)\cdot\sum_{t\in[2:s']}\tilde{\beta}_t$ and $\sum_{j'\in[2:s]}w_{s'}(j') \leq \tilde{W}\cdot\sum_{t\in[2:s']}\tilde{\beta}_t$. We have $w_{s'}(1) \geq \frac{w(1)}{\tilde{W}}\cdot\sum_{j'\in[2:s]}w_{s'}(j')$. Therefore, because $w|_{[2:s']} = \tilde{w}|_{[2:s']}$ and $W = \tilde{W} + w(1)$, we have

$$
\frac{\sum_{j\in[s]}w_{s'}(j)}{W} \geq \frac{\sum_{j\in[2:s]}w_{s'}(j)}{\tilde{W}} = \frac{\sum_{j\in[2:s]}\tilde{w}_{s'}(j)}{\tilde{W}}\,.
$$

Again, because $w|_{[2:s]} = \tilde{w}|_{[2:s]}$, we have $\mathrm{cost}_{w_{s'}}([s]) \geq \mathrm{cost}_{\tilde{w}_{s'}}([2:s])$ according to the definition of the weighted k-Means problem. We have completed the proof for this case.

The more complicated case is that $s' \geq t_0$. In this case, $w_{s'}(1) = w(1)$ because we have $w_s(1) = w(1)$; and $\eta_t(1) = 0$ for any $t > s'$

because $1 \in DC^t$. Suppose that

$$\sum_{t\in[2:t_0-1]} \tilde{B}_t = \sum_{j\in[2:s]} \tilde{w}_{t_0-1}(j)\cdot\alpha_j^* - \mathsf{PD}_1\,,$$
$$\sum_{t\in[t_0:s']} \tilde{B}_t = \sum_{j\in[2:s]} (\tilde{w}_{s'}(j)-\tilde{w}_{t_0-1}(j))\cdot\alpha_j^* - \mathsf{PD}_2\,.$$

According to our induction hypothesis, we have

$$\mathsf{PD}_1 \le \frac{c\cdot\sum_{j\in[2:s]}\tilde{w}_{t_0-1}(j)}{\tilde{W}}\cdot \mathrm{cost}_{\tilde{w}_{t_0-1}}([2:s])\,,\text{ and}$$
$$\mathsf{PD}_1+\mathsf{PD}_2 \le \frac{c\cdot\sum_{j\in[2:s]}\tilde{w}_{s'}(j)}{\tilde{W}}\cdot \mathrm{cost}_{\tilde{w}_{s'}}([2:s])\,.$$

Next, we split $\sum_{t\in[s']} B_t$ by two parts: $\sum_{t\in[t_0-1]} B_t$ and $\sum_{t\in[t_0:s']} B_t$. Note that Φ terms in the first t_0-1 terms do not change, i.e., $\Phi_j^{(t)} = \tilde{\Phi}_j^{(t)}$ for any $t\in[2:t_0-1]$ and $j\in DC^t$. For the first term, we have (assuming $\tilde{w}_1(j')=0$ for any $j'\in[s]$)

$$\begin{aligned}\sum_{t\in[t_0-1]} B_t &= B_1 + \sum_{t\in[2:t_0-1]}\left(\tilde{B}_t + \tilde{\beta}_t\cdot w(1)\cdot\alpha_1^*\right)\\ &= w(1)\cdot\alpha_1^* + \sum_{j\in[2:s]}\tilde{w}_{t_0-1}(j)\cdot\alpha_j^* - \mathsf{PD}_1\end{aligned}$$

However, for the second term, because we have an additional term $\Phi_1^{(t)}$ for each $t\ge t_0$ and we have $\beta_t = \frac{\tilde{W}}{W}\cdot\tilde{\beta}_t$ and $\phi_{1,j}^{(t)} = \frac{1}{\tilde{W}}\cdot\frac{\tilde{\eta}_t(j)}{\tilde{\beta}_t}$ for each $t\ge t_0$ and $j\in IC^t$, we need a more careful derivation:

$$\begin{aligned}&\sum_{t\in[t_0:s']} B_t\\
&= \sum_{t\in[t_0:s']}\left(\frac{\beta_t}{\tilde{\beta}_t}\cdot\tilde{B}_t + \beta_t\cdot w(1)\cdot\Phi_1^{(t)}\right)\\
&= \sum_{t\in[t_0:s']}\left(\frac{\beta_t}{\tilde{\beta}_t}\cdot\tilde{B}_t + \beta_t\cdot w(1)\cdot\sum_{j'\in IC^t}\phi_{1,j'}^{(t)}\cdot(\alpha_{j'}^* - c\cdot d^2(1,j'))\right)\\
&= \sum_{t\in[t_0:s']}\left(\frac{\tilde{W}}{W}\cdot\tilde{B}_t + \frac{\tilde{W}w(1)\tilde{\beta}_t}{W}\sum_{j'\in IC^t}\frac{\tilde{\eta}_t(j')}{\tilde{W}\tilde{\beta}_t}\cdot(\alpha_{j'}^* - c\cdot d^2(1,j'))\right)\\
&= \frac{\tilde{W}}{W}\sum_{t\in[t_0:s']}\tilde{B}_t + \frac{w(1)}{W}\sum_{t\in[t_0:s']}\sum_{j'\in IC^t}\tilde{\eta}_t(j')(\alpha_{j'}^* - c\cdot d^2(1,j'))\\
&= \frac{\tilde{W}}{W}\left(\sum_{j'\in[2:s]}(\tilde{w}_{s'}(j')-\tilde{w}_{t_0-1}(j'))\alpha_{j'}^* - \mathsf{PD}_2\right)\\
&\quad + \frac{w(1)}{W}\sum_{j'\in[2:s]}(\tilde{w}_{s'}(j')-\tilde{w}_{t_0-1}(j'))(\alpha_{j'}^* - c\cdot d^2(1,j'))\\
&= \sum_{j'\in[2:s]}(\tilde{w}_{s'}(j')-\tilde{w}_{t_0-1}(j'))\left(\alpha_{j'}^* - \frac{cw(1)d^2(1,j')}{W}\right) - \frac{\tilde{W}}{W}\cdot\mathsf{PD}_2\end{aligned}$$

Because we have $w_{s'}(j') = \tilde{w}_{s'}(j')$ for any $j'\in[2:s]$, we have

$$\begin{aligned}&\sum_{t\in[s']} B_t = \sum_{j'\in[s]} w_{s'}(j')\cdot\alpha_{j'}^* -\\
&\left(\mathsf{PD}_1 + \frac{\tilde{W}}{W}\cdot\mathsf{PD}_2 + c\sum_{j'\in[2:s]}\frac{w(1)(w_{s'}(j')-w_{t_0-1}(j'))}{W}\cdot d^2(1,j')\right)\\
&\hfill\text{(pairwise distance)}\end{aligned}$$

It suffices to upper bound the pairwise distance part. Note that the pairwise distance can be rewritten as

$$\begin{aligned}&\frac{\tilde{W}}{W}\cdot(\mathsf{PD}_1+\mathsf{PD}_2) + c\sum_{j'\in[2:s]}\frac{w(1)\cdot w_{s'}(j')}{W}\cdot d^2(1,j')\\
&\qquad + \frac{w(1)}{W}\cdot\left(\mathsf{PD}_1 - c\sum_{j'\in[2:s]} w_{t_0-1}(j')\cdot d^2(1,j')\right)\end{aligned}$$

According to our induction hypothesis,

$$\begin{aligned}\mathsf{PD}_1 &\le \frac{c\cdot\sum_{j'\in[2:s]}\tilde{w}_{t_0-1}(j')}{\tilde{W}}\cdot\mathrm{cost}_{\tilde{w}_{t_0-1}}([2:s])\\
&\le c\cdot\mathrm{cost}_{\tilde{w}_{t_0-1}}([2:s])\\
&= c\cdot\mathrm{cost}_{w_{t_0-1}}([2:s]) \qquad (w_{t_0}|_{[2:s]} = \tilde{w}_{t_0}|_{[2:s]})\\
&\le c\cdot\sum_{j'\in[2:s]} w_{t_0-1}(j')\cdot d^2(1,j')\,.\end{aligned}$$

Again, according to our induction hypothesis, we have

$$\begin{aligned}\frac{\tilde{W}}{W}\cdot(\mathsf{PD}_1+\mathsf{PD}_2) &\le \frac{c\cdot\sum_{j'\in[2:s]}\tilde{w}_{s'}(j')}{W}\cdot\mathrm{cost}_{\tilde{w}_{s'}}([2:s])\\
&= \frac{c\cdot\sum_{j'\in[2:s]} w_{s'}(j')}{W}\cdot\mathrm{cost}_{w_{s'}}([2:s])\\
&\hfill (w_{s'}|_{[2:s]} = \tilde{w}_{s'}|_{[2:s]})\\
&= \frac{c}{W}\cdot\sum_{j,j'\in\binom{[2:s]}{2}} w_{s'}(j)\cdot w_{s'}(j')\cdot d^2(j,j')\end{aligned}$$

Because $w_{s'}(1) = w(1)$, we have

$$\begin{aligned}&\frac{\tilde{W}}{W}\cdot(\mathsf{PD}_1+\mathsf{PD}_2) + c\cdot\sum_{j'\in[2:s]}\frac{w(1)\cdot w_{s'}(j')}{W}\cdot d^2(1,j')\\
\le&\frac{c}{W}\cdot\sum_{j,j'\in\binom{[s]}{2}} w_{s'}(j)\cdot w_{s'}(j')\cdot d^2(j,j')\\
=&\frac{c\cdot\sum_{j'\in[s]} w_{s'}(j')}{W}\cdot\mathrm{cost}_{w_{s'}}([s])\,.\end{aligned}$$

Therefore, the pairwise distance part can be upper bounded by $\frac{c\cdot\sum_{j'\in[s]} w_{s'}(j')}{W}\cdot\mathrm{cost}_{w_{s'}}([s])$. In conclusion, we get

$$\sum_{t\in[s']} B_t \ge \sum_{j'\in[s]} w_{s'}(j')\cdot\alpha_j^* - \frac{c\cdot\sum_{j'\in[s]} w_{s'}(j')}{W}\cdot\mathrm{cost}_{w_{s'}}([s])\,.$$

Bullet E of Lemma 13. If $1\in IC^s$, we have $\phi_{j,1}^{(s)} = 0$ for any $j\in DC^s$ and $\phi_{j,j'}^{(s)} = \tilde{\phi}_{j,j'}^{(s)}$ for any $j\in DC^s\setminus$ and $j'\in IC^s\setminus\{1\}$. Therefore, Bullet E follows the induction hypothesis directly.

If $1\in DC^s$, according to our definition, for any $j'\in IC^s$, we have $\phi_{1,j'}^{(s)} = \frac{1}{\tilde{\beta}_s}\cdot(w(j') + \sum_{j\in DC^s\setminus\{1\}} w(j)\cdot\tilde{\phi}_{j,j'}^{(s)})$. Because $\tilde{\phi}_{j,j'}^{(s)}/w(j')$ is monotone in j', we have $\phi_{1,j'}^{(s)}/w(j')$ is also monotone in j'. The monotonicity of $\phi_{j,j'}^{(s)}/w(j')$ for any $j\in DC^s\setminus\{1\}$ follows the definition that $\phi_{j,j'}^{(s)} = \tilde{\phi}_{j,j'}^{(s)}$ and the induction hypothesis directly.

Acknowledgments

Ruiquan Gao was supported by a Stanford Gradudate Fellowship, NSF CCF-2112824, and a David and Lucile Packard Fellowship. Fabrizio Grandoni was partially supported by the SNF Grants 200021-200731 and 200021-236706. Euiwoong Lee was supported in part by NSF grant 2236669. Ernest van Wijland was supported in part by the French PEPR integrated projects EPIQ (ANR-22-PETQ-0007).

References

[1] Sara Ahmadian, Ashkan Norouzi-Fard, Ola Svensson, and Justin Ward. 2020. Better Guarantees for k-Means and Euclidean k-Median by Primal-Dual Algorithms. *SIAM J. Comput.* 49, 4 (2020). https://doi.org/10.1137/18M1171321

[2] Vijay Arya, Naveen Garg, Rohit Khandekar, Adam Meyerson, Kamesh Munagala, and Vinayaka Pandit. 2004. Local Search Heuristics for k-Median and Facility Location Problems. *SIAM J. Comput.* 33, 3 (2004), 544–562. https://doi.org/10.1137/S0097539702416402

[3] Pranjal Awasthi, Moses Charikar, Ravishankar Krishnaswamy, and Ali Kemal Sinop. 2015. The hardness of approximation of euclidean k-means. *arXiv preprint arXiv:1502.03316* (2015).

[4] Jaroslaw Byrka and Karen Aardal. 2010. An Optimal Bifactor Approximation Algorithm for the Metric Uncapacitated Facility Location Problem. *SIAM J. Comput.* 39, 6 (2010), 2212–2231. https://doi.org/10.1137/070708901

[5] Jaroslaw Byrka, Thomas W. Pensyl, Bartosz Rybicki, Aravind Srinivasan, and Khoa Trinh. 2017. An Improved Approximation for k-Median and Positive Correlation in Budgeted Optimization. *ACM Trans. Algorithms* 13, 2 (2017), 23:1–23:31. https://doi.org/10.1145/2981561

[6] Moses Charikar, Vincent Cohen-Addad, Ruiquan Gao, Fabrizio Grandoni, Euiwoong Le, and Ernest Van Wijland. 2025. An Improved Greedy Approximation for (Metric) k-Means. In *2025 IEEE 66th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 233–240.

[7] Moses Charikar and Sudipto Guha. 2005. Improved Combinatorial Algorithms for Facility Location Problems. *SIAM J. Comput.* 34, 4 (2005), 803–824. https://doi.org/10.1137/S0097539701398594

[8] Moses Charikar, Sudipto Guha, Éva Tardos, and David B. Shmoys. 1999. A Constant-Factor Approximation Algorithm for the k-Median Problem (Extended Abstract). In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, May 1-4, 1999, Atlanta, Georgia, USA*, Jeffrey Scott Vitter, Lawrence L. Larmore, and Frank Thomson Leighton (Eds.). ACM, 1–10. https://doi.org/10.1145/301250.301257

[9] Fabián A Chudak and David P Williamson. 1999. Improved approximation algorithms for capacitated facility location problems. In *International Conference on Integer Programming and Combinatorial Optimization*. Springer, 99–113.

[10] Vincent Cohen-Addad, Hossein Esfandiari, Vahab S. Mirrokni, and Shyam Narayanan. 2022. Improved approximations for Euclidean k-means and k-median, via nested quasi-independent sets. In *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, Stefano Leonardi and Anupam Gupta (Eds.). ACM, 1621–1628. https://doi.org/10.1145/3519935.3520011

[11] Vincent Cohen-Addad, Andreas Emil Feldmann, and David Saulpic. 2021. Near-linear time approximation schemes for clustering in doubling metrics. *Journal of the ACM (JACM)* 68, 6 (2021), 1–34.

[12] Vincent Cohen-Addad, Fabrizio Grandoni, Euiwoong Lee, and Chris Schwiegelshohn. 2023. Breaching the 2 LMP Approximation Barrier for Facility Location with Applications to k-Median. In *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, Nikhil Bansal and Viswanath Nagarajan (Eds.). SIAM, 940–986. https://doi.org/10.1137/1.9781611977554.CH37

[13] Vincent Cohen-Addad, Fabrizio Grandoni, Euiwoong Lee, Chris Schwiegelshohn, and Ola Svensson. 2025. A $(2+\epsilon)$-Approximation Algorithm for Metric k-Median. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing, STOC 2025, Prague, Czechia, June 23-27, 2025*, Michal Koucký and Nikhil Bansal (Eds.). ACM, 615–624. https://doi.org/10.1145/3717823.3718299

[14] Vincent Cohen-Addad, Karthik C S, and Euiwoong Lee. 2022. Johnson coverage hypothesis: Inapproximability of k-means and k-median in ℓp-metrics. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 1493–1530.

[15] Sanjoy Dasgupta. 2008. The hardness of k-means clustering. (2008).

[16] W Fernandez De La Vega, Marek Karpinski, Claire Kenyon, and Yuval Rabani. 2003. Approximation schemes for clustering problems. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*. 50–58.

[17] Dan Feldman, Morteza Monemizadeh, and Christian Sohler. 2007. A PTAS for k-means clustering based on weak coresets. In *Proceedings of the twenty-third annual symposium on Computational geometry*. 11–18.

[18] Kishen N. Gowda, Thomas W. Pensyl, Aravind Srinivasan, and Khoa Trinh. 2023. Improved Bi-point Rounding Algorithms and a Golden Barrier for k-Median. In *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, Nikhil Bansal and Viswanath Nagarajan (Eds.). SIAM, 987–1011. https://doi.org/10.1137/1.9781611977554.CH38

[19] Fabrizio Grandoni, Rafail Ostrovsky, Yuval Rabani, Leonard J. Schulman, and Rakesh Venkat. 2022. A refined approximation for Euclidean k-means. *Inf. Process. Lett.* 176 (2022), 106251. https://doi.org/10.1016/J.IPL.2022.106251

[20] Sudipto Guha and Samir Khuller. 1999. Greedy Strikes Back: Improved Facility Location Algorithms. *J. Algorithms* 31, 1 (1999), 228–248. https://doi.org/10.1006/jagm.1998.0993

[21] Mary Inaba, Naoki Katoh, and Hiroshi Imai. 1994. Applications of weighted Voronoi diagrams and randomization to variance-based k-clustering. In *Proceedings of the tenth annual symposium on Computational geometry*. 332–339.

[22] Kamal Jain, Mohammad Mahdian, Evangelos Markakis, Amin Saberi, and Vijay V. Vazirani. 2003. Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *J. ACM* 50, 6 (2003), 795–824. https://doi.org/10.1145/950620.950621

[23] Kamal Jain, Mohammad Mahdian, and Amin Saberi. 2002. A new greedy approach for facility location problems. In *Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada*. 731–740. https://doi.org/10.1145/509907.510012

[24] K. Jain and V. Vazirani. 2001. Approximation algorithms for metric facility location and k-Median problems using the primal-dual schema and Lagrangian relaxation. *J. ACM* 48, 2 (2001), 274–296. https://doi.org/10.1145/375827.375845

[25] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu. 2004. A local search approximation algorithm for k-means clustering. *Comput. Geom.* 28, 2-3 (2004), 89–112. https://doi.org/10.1016/j.comgeo.2004.03.003

[26] Amit Kumar, Yogish Sabharwal, and Sandeep Sen. 2010. Linear-time approximation schemes for clustering problems in any dimensions. *J. ACM* 57, 2 (2010), 5:1–5:32. https://doi.org/10.1145/1667053.1667054

[27] Euiwoong Lee, Melanie Schmidt, and John Wright. 2017. Improved and simplified inapproximability for k-means. *Inform. Process. Lett.* 120 (2017), 40–43.

[28] Euiwoong Lee and Kijun Shin. 2025. Facility Location on High-Dimensional Euclidean Spaces. In *16th Innovations in Theoretical Computer Science Conference (ITCS 2025)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 70–1.

[29] Shi Li. 2013. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Inf. Comput.* 222 (2013), 45–58. https://doi.org/10.1016/j.ic.2012.01.007

[30] Shi Li and Ola Svensson. 2016. Approximating k-Median via Pseudo-Approximation. *SIAM J. Comput.* 45, 2 (2016), 530–547. https://doi.org/10.1137/130938645

[31] S. P. Lloyd. 1957. Least squares quantization in PCM. *Bell Telephone Laboratories Memo* (1957). Unpublished; often cited for the k-means algorithm.

[32] J. B. MacQueen. 1967. Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1*, L. M. Le Cam and J. Neyman (Eds.). University of California Press, Berkeley, CA, 281–297.

[33] Mohammad Mahdian, Yinyu Ye, and Jiawei Zhang. 2006. Approximation Algorithms for Metric Facility Location Problems. *SIAM J. Comput.* 36, 2 (2006), 411–432. https://doi.org/10.1137/S0097539703435716

[34] Jiří Matoušek. 2000. On approximate geometric k-clustering. *Discrete & Computational Geometry* 24, 1 (2000), 61–84.

[35] David B. Shmoys, Éva Tardos, and Karen Aardal. 1997. Approximation Algorithms for Facility Location Problems (Extended Abstract). In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*, Frank Thomson Leighton and Peter W. Shor (Eds.). ACM, 265–274. https://doi.org/10.1145/258533.258600

[36] Hugo Steinhaus. 1957. Sur la division des corps matériels en parties. *Bulletin de l'Académie Polonaise des Sciences, Classe III* 4, 12 (1957), 801–804. In French.

Received 2025-11-04; accepted 2026-02-01