A Nearly Tight Analysis of Greedy k-means++

Christoph Grunau
cgrunau@inf.ethz.ch
ETH Zurich

Ahmet Alper Özüdoğru
oahmet@student.ethz.ch
ETH Zurich

Václav Rozhoň
rozhonv@inf.ethz.ch
ETH Zurich

Jakub Tětek
j.tetek@gmail.com
BARC, Univ. of Copenhagen

Abstract

The famous k-means++ algorithm of Arthur and Vassilvitskii [SODA 2007] is the most popular way of solving the k-means problem in practice. The algorithm is very simple: it samples the first center uniformly at random and each of the following $k-1$ centers is then always sampled proportional to its squared distance to the closest center so far. Afterward, Lloyd's iterative algorithm is run. The k-means++ algorithm is known to return a $\Theta(\log k)$ approximate solution in expectation.

In their seminal work, Arthur and Vassilvitskii [SODA 2007] asked about the guarantees for its following *greedy* variant: in every step, we sample ℓ candidate centers instead of one and then pick the one that minimizes the new cost. This is also how k-means++ is implemented in e.g. the popular Scikit-learn library [Pedregosa et al.; JMLR 2011].

We present nearly matching lower and upper bounds for the greedy k-means++: We prove that it is an $O(\ell^3 \log^3 k)$-approximation algorithm. On the other hand, we prove a lower bound of $\Omega(\ell^3 \log^3 k / \log^2(\ell \log k))$. Previously, only an $\Omega(\ell \log k)$ lower bound was known [Bhattacharya, Eube, Röglin, Schmidt; ESA 2020] and there was no known upper bound.

Contents

1 Introduction

This paper is devoted to analyzing a natural and frequently-used greedy variant of the famous k-means++ clustering algorithm [AV07]. The difference between k-means++ and its greedy variant is very small: k-means++ samples one center in each step while greedy k-means++ samples ℓ candidate centers and then selects the one that decreases the current cost the most. While it is well known that the k-means++ algorithm is $\Theta(\log k)$-approximate, analyzing its greedy variant remained wide open. In this paper we show that greedy k-means++ is $O(\ell^3 \log^3 k)$-approximate. Suprisingly, this is nearly tight. Specifically, we prove a lower bound of $\Omega(\ell^3 \log^3 k / \log^2(\ell \log k))$.

Clustering: Clustering is one of the most important tools in unsupervised machine learning. The task is to divide given input data into clusters of neighboring data points. There are many ways of formalizing that task, but one of the most popular ones is the k-means problem.

In the k-means problem, we are given a set of points $X \subseteq \mathbb{R}^d$, as well as a parameter k. We are asked to find a set of k *centers* $K \subseteq \mathbb{R}^d$ that minimizes the sum of squared distances of points of X to their respective closest centers. Namely, if we define the *cost* of a point x with respect to a set of centers C as $\varphi(x, C) := \min_{c \in C} ||x - c||_2^2$, we wish to find a set C with $|C| = k$ that minimizes the expression $\varphi(X, C) := \sum_{x \in X} \varphi(x, C)$.

The k-means problem is NP-hard [ADHP09, MNV09] and also NP-hard to approximate within some constant factor $c > 1$ [ACKS15, LSW17]. On the other hand, there is a long line of work on approximation algorithms, with the current record holder being the algorithm of [CAEMN22] with approximation ratio of 5.912. Moreover, $1 + \varepsilon$-approximation can be reached for constant d [FMS07] and constant k [KSS04].

However, these algorithms are not used in practice. Instead, practitioners rely on the so-called Lloyd's heuristic [Llo82] which can start with an arbitrary solution and iteratively makes its cost smaller, until convergence.

Lloyd's heuristic is not ideal: it is prone to get stuck in bad local optima [AV07]. In particular, it is not a constant approximation algorithm. A remedy to this problem is seeding it with a solution that is already good as the Lloyd's heuristic can then only make its cost smaller. In practice, such a seed can be simply a random subset of X of size k. This natural option is for instance one of the options in the implementation used by Scikit learn [PVG+11] or R [R C13]. Such an approach does not lead to any approximation ratio guarantees; its approximation ratio can be arbitrarily bad in some simple instances, e.g. whenever we have k well-separated clusters lying in one line.

k**-means++**: A major result of Arthur and Vassilvitskii [AV07, ORSS13] is a simple seeding algorithm known as k-means++ that both works well in practice and has desirable theoretical worst-case guarantees.

The k-means++ algorithm works as follows. We sample the set $C \subseteq X$ sequentially, one center at a time. The first center we sample as a uniform point from X. Each next center is sampled proportional to its current cost. That is, if C_i is the already constructed set of centers, we sample $x \in X$ as the next one with probability $\varphi(x, C_i)/\varphi(X, C_i)$. The pseudocode is in Algorithm 1.

Algorithm 1 k-means++ seeding

Input: X, k

1: Uniformly sample $x \in X$ and set $C_1 = \{x\}$.
2: **for** $i \leftarrow 1, 2, 3, \ldots, k - 1$ **do**
3:     Sample $x \in X$ w.p. $\frac{\varphi(x, C_i)}{\varphi(X, C_i)}$ and set $C_{i+1} = C_i \cup \{x\}$.
4: **return** $C := C_k$

Arthur and Vassilvitskii proved that Algorithm 1 is $\Theta(\log k)$ approximate, in expectation[1]. Although this approximation guarantee is not even constant, a benchmark achievable by many other known polynomial-time algorithms [KMN+04, LS19, ANFSW19], the main point of the analysis is that we cannot construct an adversarial instance where the Lloyd's heuristic seeded by k-means++ seeding can be arbitrarily bad, as is the case for e.g. the uniform random seeding. On practical data sets, the k-means++ seeding gives consistently better results than the random seeding [AV07] and is implemented in popular machine learning libraries like Scikit-learn [PVG+11].

However, it turns out that the algorithm implemented in the popular Scikit-learn library is *not* the basic k-means++ (Algorithm 1), but its *greedy* variant described in Algorithm 2. This algorithm in fact comes from the original paper of Arthur and Vassilvitskii [AV07] who mention that it gives better empirical results. They say that their analysis "do[es] not carry over to this scenario" and that "it would be interesting to see a comparable (or better) asymptotic result".

The greedy k-means++ algorithm works as follows. In every step, we sample ℓ *candidate centers* $c_{i+1}^1, \ldots, c_{i+1}^\ell$ from the constructed distribution, not just one. Next, for each candidate center c_{i+1}^j we compute the new cost of the solution $\varphi(X, C \cup \{c_{i+1}^j\})$ if we add this candidate to our set of centers. Then we pick the candidate center that minimizes this expression.

Algorithm 2 Greedy k-means++ seeding

Input: X, k, ℓ

1: Uniformly independently sample $c_1^1, \ldots, c_1^\ell \in X$;
2: Let $c_1 = \arg\min_{c \in \{c_1^1, \ldots, c_1^\ell\}} \varphi(X, c)$ and set $C_1 = \{c_1\}$.
3: **for** $i \leftarrow 1, 2, 3, \ldots, k-1$ **do**
4:     Sample $c_{i+1}^1, \ldots, c_{i+1}^\ell \in X$ independently, sampling x with probability $\frac{\varphi(x, C_i)}{\varphi(X, C_i)}$;
5:     Let $c_{i+1} = \arg\min_{c \in \{c_i^1, \ldots, c_i^\ell\}} \varphi(X, C_i \cup \{c\})$ and set $C_{i+1} = C_i \cup \{c_{i+1}\}$.
6: **return** $C := C_k$

In [BERS20], the authors show that Algorithm 2 is $\Omega(\ell \log k)$ approximate, in expectation. That is, the enhanced Algorithm 2 has worse theoretical guarantees than the original Algorithm 1! This should not, however, be so surprising. In fact, if we think about ℓ going to infinity, the algorithm becomes essentially deterministic; in every step, it will simply pick the point that decreases the cost the most. Such an algorithm heuristically makes sense, but lacks worst-case guarantees[2]. To see this, imagine two clusters with many points and a single lonely point in between: taking the lonely point results in a substantial drop of the cost, but it cannot be part of any solution with bounded approximation factor.

So, we cannot hope that Algorithm 2 gets better guarantees than Algorithm 1. However, in the words of Arthur and Vassilvitskii, its guarantees could still be "comparable". This is the main result of this paper:

Theorem 1.1. *Greedy k-means++ (Algorithm 2) is an $O(\ell^3 \cdot \log^3 k)$-approximation algorithm, in expectation.*

On the other hand, we provide the following near-matching lower bound.

Theorem 1.2. *For every k and $\ell \leq k^{0.1}$, there exists a point set $X \subseteq \mathbb{R}^d$ for some $d \in \mathbb{N}$ where Algorithm 2 outputs $\Omega(\ell^3 \log^3 k / \log^2(\ell \log k))$ approximate solution with constant probability.*

---

[1]In practice, the algorithm is rerun several times, which boosts the guarantee in expectation to a high probability guarantee.

[2]Note that the negative cost $-\varphi(X, C)$ is submodular in C, but because of the negative sign we cannot use the well-known fact that the greedy algorithm yields a $1 - 1/e$ approximation of the optimum.

We believe that the $\widetilde{\Theta}(\ell^3 \log^3 k)$-approximation[3] bound is a truly unexpected twist! However bizarre it may sound now, we hope to give an adequate intuition behind it in Section 2.

Greedy rule is crucial: The second result of this paper is that the greedy heuristic in Algorithm 2 is in fact crucial to getting polylogarithmic approximation. To this end, we generalize Algorithm 2 by allowing each center to be chosen from ℓ candidates by an arbitrary rule. The approximation ratio of this general algorithm becomes polynomial in k, unless we know the specifics of the rule. Concretely, we prove:

Theorem 1.3 (Informal version of Theorem A.1). *There exists a point set $X \subseteq \mathbb{R}^d$ and a rule $\mathcal{R}$ such that a variant of Algorithm 2 that uses $\mathcal{R}$ instead of the greedy rule is $\Omega(k^{1-1/\ell})$-approximate with constant probability.*

This theorem also suggests that the greedy heuristic, among all others, makes a lot of sense! On the other hand, we get the following upper bound.

Theorem 1.4 (Informal version of Theorem B.1). *For any rule $\mathcal{R}$, a variant of Algorithm 2 that uses $\mathcal{R}$ instead of the greedy rule is $O(k^{2-1/\ell} \cdot \ell \log k)$-approximate.*

We note that from the perspective of Algorithms with Predictions [MV20], Theorem 1.4 shows that whatever rule we use to generalize the greedy k-means++ algorithm, we still know that the algorithm remains somewhat comparable to the optimum solution.

The gap between the lower bound and upper bound of Theorems 1.3 and 1.4 can be tightened if one analyzes a certain natural stochastic process that can be understood without knowing anything about k-means($++$); we defer the discussion of this interesting open problem to Appendix B.2.

Related work: In this section, we list some related work with algorithms derived from k-means++. To see more work about k-means in general, see for example the introduction of [CAEMN22].

To the best of our knowledge, the only paper exploring Algorithm 2 from the theoretical perspective after the seminal paper [AV07] of Arthur and Vassilvitskii, is the paper of Bhattacharya, Eube, Röglin, and Schmidt [BERS20]. There, the authors show an $\Omega(\ell \log k)$ lower bound. They also consider a problem closely related to analysis of Algorithm 3, we discuss it in greater detail in Appendix B.

The empirical work related to Algorithm 2 starts with the PhD thesis of Vassilvitskii [Vas07] that reports experiments with $\ell = 2$, the comparative study of [CKV13] on the other hand advertises the choice of $\Theta(\log k)$. This is also the choice taken in the Scikit-learn implementation [PVG$^+$11] that chooses $\ell = \lfloor 2 + \ln k \rfloor$.

Another related work to k-means++ include the following. Lattanzi and Sohler [LS19, CGPR20] present a variant of k-means++ inspired by the local search algorithm of [KMN$^+$04]. A popular distributed variant of the k-means++ algorithm is the k-means$\|$ algorithm of Bahmani, Moseley, Vattani, Kumar, and Vassilvitskii [BMV$^+$12, BLK17, Roz20, MRS20] that achieves the same guarantees as k-means++ in $O(\log n)$ Map-Reduce rounds. Other lines of work study bicriteria guarantees of k-means++ [ADK09, Wei16, MRS20], analyze bad instances [BR13], speed-up k-means++ by subsampling [BLHK16b, BLHK16a] or adapt it to the setting with outliers [BVX19, GR20].

Roadmap: In Section 2, we explain intuitively the proofs of Theorems 1.1, 1.2, A.1 and B.1. Section 3 collects some basic preliminary results that we need to use. In Section 4 we prove the main result necessary to prove Theorem 1.1 which is then proved in Section 5. In Section 6 we then construct the almost matching lower bound. The analysis of Algorithm 3 is deferred to Appendices A and B.

---

[3]Throughout this paper, we use $\widetilde{O}(f(x))$ to denote $O(f(x) \operatorname{poly} \log(f(x)))$ and $\widetilde{\Omega}, \widetilde{\Theta}$ are defined analogously.

2 Intuitive explanations

This section is devoted to an intuitive explanation of Theorems 1.1, 1.2, A.1 and B.1. We start by reviewing the analysis of the k-means++ algorithm by [AV07] in Section 2.1. Next, in Section 2.2 we identify the issues with generalizing the analysis of k-means++ to its greedy variant from Algorithm 2. In Section 2.3, we discuss a crucial lemma that essentially says that the greedy rule implies that not so many candidate centers are sampled in total from each optimal cluster. In Section 2.4, we show how this lemma implies Theorem 1.1. Finally, we present the lower bound in Section 2.5.

2.1 Analyzing k-means++

We need to start by reviewing the analysis of the k-means++ algorithm by Arthur and Vassilvitskii. Reviewing it will allow us to explain which parts of the argument cannot be simply generalized in the analysis of Algorithm 2.

For $K \subseteq X$ we define $\varphi^*(K) = \varphi(K, \mu(K))$ where $\mu(K) = (\sum_{c \in K} c)/|K|$ is the center of mass of K. We note that $\varphi^*(K)$ is the optimal k-means cost of K achievable with one center. At the core of the k-means++ analysis lies the following lemma.

Lemma 2.1 (Informal version of Lemma 3.3). *Let $X \subseteq \mathbb{R}^d$ be the input point set to the k-means problem and $C \subseteq \mathbb{R}^d$ a set of already selected centers. Let K be an arbitrary subset of X. If we sample a point of K such that a point $c \in K$ is sampled with probability $\varphi(c, C)/\varphi(K, C)$, we have $E[\varphi(K, C \cup \{c\})] \leq 5\varphi^*(K)$.*

Intuitively, the reason why the lemma holds is that either all points C are far away from $\mu(K)$ and then we essentially sample $c \in K$ from a uniform distribution; such a distribution would even lead to $E[\varphi(K, \{c\})] = 2\varphi^*(K)$ by a simple averaging argument (cf. Lemma 3.2). The other option is that some point of C is already close to $\mu(K)$, but then $\varphi(K, C)$ is already small.

Whenever we use this lemma, we have K to be a *cluster* of some fixed optimal solution. Here, given a set of centers C, we define a cluster $K \subseteq X$ of a center $c \in C$ as the set of the points for which c is the closest center, that is, $K = \{x \in X : c = \mathrm{argmin}_{c' \in C} \varphi(x, c')\}$. The usefulness of Lemma 2.1 comes from its corollary that whenever Algorithm 1 samples the new center c_i in the i-th step and it happens that $c_i \in K$, then the cost of K becomes 5-approximated, in expectation. This holds even though the algorithm itself has no idea about what the optimal clusters are.

So, if it somehow happened that each of the k sampled centers was from a different optimal cluster, we would get that Algorithm 1 is a 5-approximation. Of course, there can be some clusters with no sampled center in the end because we happened to hit some other optimal cluster twice or more. This is the reason behind the final $O(\log k)$ approximation.

Let us be more precise now: Given a set of already taken centers C_i, we say that a cluster K from the optimal solution is *covered* in the $i + 1$th step if $K \cap C_i \neq \emptyset$. We accordingly split X into points in covered and uncovered clusters, i.e., $X = X_i^{\mathcal{C}} \sqcup X_i^{\mathcal{U}}$.[4] Finally, we say that a step $i + 1$ is *good* whenever c_{i+1} is sampled from an uncovered cluster and *bad* otherwise. It is the bad steps, where the algorithm really loses ground with respect to the optimal solution. The question is, by how much?

Intuitively, if there are u_i uncovered clusters remaining after i steps and we had a bad $i + 1$th step, in the end we will need to pay in our solution for one of the u_i currently uncovered clusters. The cost of the largest currently uncovered cluster may be as large as $\varphi(X_i^{\mathcal{U}}, C_i)$. However, we should hope that we only need to pay the cost of the average one, i.e., $\varphi(X_i^{\mathcal{U}}, C_i)/u_i$. The reason

---

[4]The notation $A = B \sqcup C$ means $A = B \cup C$ and $B \cap C = \emptyset$.

for that is that we expect the size of the average uncovered cluster to only go down in the future, i.e., we have $\mathrm{E}\left[\varphi(X_j^{\mathcal{U}}, C_j)/u_j\right] \le \varphi(X_i^{\mathcal{U}}, C_i)/u_i$ for $j > i$. This is because if each new covered cluster were chosen from uncovered ones uniformly at random, the average cost of an uncovered cluster would stay the same. However, we are actually covering the more costly clusters with higher probability which makes the expected average cost of an uncovered cluster decrease in the future steps.

So, having a bad step incurs a cost of $\varphi(X_i^{\mathcal{U}}, C_i)/u_i$. This cost can be huge but, very conveniently, in that case, the probability of the $i+1$-th step being bad was very small to begin with. More concretely, the probability of having a bad step is equal to $\varphi(X_i^{\mathcal{C}}, C_i)/\varphi(X, C_i)$. Since the numerator of that expression is in expectation at most $5OPT$ by Lemma 2.1, we conclude that the expected cost incurred by the fact that we may sample from an already covered cluster in step $i+1$ is at most

$$\frac{5OPT}{\varphi(X, C_i)} \cdot \frac{\varphi(X_i^{\mathcal{U}}, C_i)}{u_i} \le \frac{5OPT}{u_i}.$$

The variable u_i starts at $u_0 = k$ and then decreases in each step by at most one until $u_{k-1} \ge 1$ in the kth step. Thus the contribution to the cost over all k steps can be upper bounded by $O(OPT) \cdot (\frac{1}{k} + \frac{1}{k-1} + \cdots + \frac{1}{1}) = O(OPT \cdot \log k)$.

This concludes the intuitive analysis of Algorithm 1. The formal proof can be found in [AV07] and a more detailed exposition can be found in lecture notes of Dasgupta [Das19].

2.2 k-means++ strikes back

Here is the main problem with generalizing the k-means++ analysis to Algorithm 2: We can sample many centers from the same cluster K of the optimal solution, before the greedy heuristic decides to pick one! In other words: in k-means++ we can simply use Lemma 2.1 to conclude that whenever we sample a point from an optimal cluster K, we expect its cost to become a 5-approximation of the optimal cost. But what if we sample a candidate center from K in every step of the algorithm and the greedy heuristic chooses to actually pick the center only when it results in a bad approximation for K? In general, we can guarantee only expected $5 \cdot (k\ell)$ approximation for the cluster K, when a center from it is picked, instead of 5 approximation.

In fact, let us now turn this idea into a lower bound. We will now show that with $\ell = \Omega(\log k)$ there is a rule $\mathcal{R}$ such that Algorithm 3 has approximation factor of $\Omega(k)$, with constant probability. A simple generalization of the following argument will then yield Theorem A.1 in Appendix A.

The weighted point set X that we use for the lower bound is in Fig. 1. We can use integer weights, even though they are not part of the original problem formulation, by replacing an element with weight w by w copies with unit weight. We will explain the lower bound in the setup where the points of X are endowed with an arbitrary metric, not necessarily the Euclidean one. Generalizations to the Euclidean space are routine and left to the formal proofs.

We start with a point d in the center of the picture that has weight much larger than all the other points combined; thus with constant probability at least one candidate center is d in the first step and our rule will then select d as the first center.

In a distance k around d, there are $k-2$ dummy points $x_1, \ldots, x_{k-2}$, each with weight one. Moreover, there is an additional pair of points c and b, with c having weight k and b having weight 1. Their distance from each other is 1 and the distance of c from d is k. The set $X = \{d, x_1, \ldots, x_{k-2}, b, c\}$ is endowed with a tree metric generated by the defined distance (see Fig. 1).

Let us compute the optimal cost of this instance: The optimum solution would take as centers all points except of b. Its cost is hence $w(b) \cdot d(b, c)^2 = 1$.
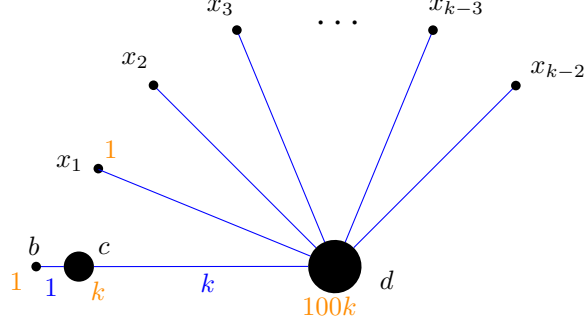
Figure 1: Illustration for the lower bound in the adversarial case. Edge weights defining the metric are in blue while vertex weights are in orange and heavier points have a larger disk. All nodes x_i have weight 1. Consider a rule that does not want to take the point c as a center unless it has to, but it wants to take b as a center whenever it can. Such a rule takes b a center with constant probability which results in $\Omega(k)$ approximation. Notice that in this case $K = \{b, c\}$ is an optimal cluster such that we sample a candidate center from it $\Omega(k)$ times but select each candidate as a center only when it results in a bad approximation of the cost of K.

Now, let us choose the rule $\mathcal{R}$ in Algorithm 3 as follows: whenever d or b is sampled as a candidate center, we select it. On the other hand, we do not select c as a center unless we have to (which happens only when all candidate centers sampled are c).

We can see that with this rule, with high probability, we select d in the very first step and then in each of the following $k/2$ steps we have only negligible probability of adding c to the set of centers: For that to happen, we would need all candidate centers to be c which happens with probability at most $\left(\frac{w(c)}{w(c)+k/2\cdot w(x_1)}\right)^\ell = (2/3)^\ell < 1/\text{poly}(k)$ by our assumption $\ell = \Omega(\log k)$. On the other hand, the probability of sampling b and thus adding it to the set of centers is $\Omega(1/k)$ in every step (unless b or c is selected as a center), hence it is constant after $k/2$ steps.

In this case, the solution of the algorithm has to leave some other point from $\{x_1, \dots, x_{k-2}, c\}$ from the set of centers. If it leaves some x_i, it needs to pay $w(x_i) \cdot d(x_i, d)^2 = k^2$. If it leaves c, it needs to pay $w(c) \cdot d(b, c)^2 = k$. That is, the solution of Algorithm 3 is k-approximate or worse with constant probability.

A more careful analysis in Theorem A.1 shows that for smaller ℓ, we should choose the weight of c to be $k^{1-1/\ell}$, to get a $\Omega(k^{1-1/\ell})$ lower bound.

2.3 A new hope

Let us observe that greedy k-means++ would not be fooled by the example from the previous Section 2.2. Once the point c is sampled as a candidate center, it is also selected as a center by the greedy rule. This is because it results in a bigger drop in the cost than if we picked any of the dummy points x_j (we assume d is already picked).

So, we may hope that in greedy k-means++ each optimal cluster cannot be hit by many candidate centers before it becomes covered. Our main technical contribution towards Theorem 1.1 is indeed a proof that each optimal cluster K is not hit by many candidate centers by greedy k-means++, before it becomes covered or before its cost is comparable with the optimal one.

Recall that a cluster K is covered in the i-th step if $K \cap C_i \neq \emptyset$. We additionally say that K is solved if $\varphi(K, C_i) \leq 10^5 \varphi^*(K)$. Finally, we define $\text{HIT}(K)$ to be the count of candidate centers $c_i^j \in K$ for all $1 \leq i \leq k$ and $1 \leq j \leq \ell$ where we count a hit only when K is not covered or solved in the respective step. Our result is then the following.

6

Lemma 2.2. *For any optimal cluster K we have $E[HIT(K)] = O(\ell^2 \log^2(k))$.*

Going back to our issue with generalizing the analysis of k-means++, we note that the above lemma implies that although we can no longer say that the cost of a cluster that just became covered is 5 approximated in expectation, we can at least expect it to be $5 \cdot O(\ell^2 \log^2 k)$ approximated. This implies that the final approximation guarantee picks up an additional $O(\ell^2 \log^2 k)$ factor. This almost explains the final $O(\ell^3 \log^3 k)$ guarantee that we achieve in Theorem 1.1 – we pick up the remaining ℓ factor inside the main analysis, essentially because the probability of having a bad step increases by a factor of ℓ. In the rest of this section, we sketch the proof of Lemma 2.2.

An important measure of progress in our analysis is the size of the *neighborhood* of K. Given a point set C_i, we define $R_i = d(\mu(K), C_i)$ and define the neighborhood N_i of a cluster K in the $i+1$-th step as the set of points closer to $\mu(K)$ than R_i (see Fig. 2). Since we assume K is not solved, i.e., that the current cost $\varphi(K, C_i)$ of K is at least $10^5 \varphi^*(K)$, we know that the distance R_i is much larger than the distance of an average point of K from $\mu(K)$. This means that most of the points of K have to lie in N_i. For simplicity, we will next assume that $K \subseteq N_i$.

We will also, for the sake of simplicity, assume that every point in N_i has distance at most $R_i/10$ from $\mu(K)$. One reason this assumption makes our life easier is that every point in N_i has cost between $(9R_i/10)^2$ and $(11R_i/10)^2$. That is, up to small factors we can think of all points of N_i having the cost R_i^2.

In every step of the algorithm, we say that we are either in the *easy* case or in the *hard* case. We say that we are in the easy case if it holds that for c sampled proportionally to its cost we have $\varphi(X, C_i) - \varphi(X, C_i \cup \{c\}) \le \varphi(N_i, C_i)/2$ with probability at least $1 - 1/\ell$, otherwise we are in the hard case. In the following discussions, we explain what needs to be done if all steps are easy and then if all the steps are hard. The fact that some steps are easy and some are hard does not make the final analysis more complicated.

Easy case: The reason why the easy case is in fact easy is that in that case, we can verify that whenever we sample a candidate center from N_i, the greedy rule adds it to the set of centers with constant probability. This intuitively means that for every hit of K we are getting a lot of progress in terms of the size of N_i going down rapidly.

More precisely, we first observe that for a fixed $1 \le j \le \ell$, whenever $c_{i+1}^j \in N_i$, we have $\varphi(X, C_i) - \varphi(X, C_i \cup \{c_{i+1}^j\}) \ge \varphi(N_i, C_i)/2$. This inequality holds because of our simplifying assumption that all points in N_i have distance at most $R_i/10$ from $\mu(K)$: this assumption implies that the cost of each point x in N_i drops from $d(x, C_i)^2 \ge (9R_i/10)^2$ to at most $(d(c_{i+1}^j, \mu(K)) + d(\mu(K), x))^2 \le (2R_i/10)^2$.

This means that whenever we sample a candidate center $c_{i+1}^j \in N_i$ and we are in the easy case, we also have constant probability of $(1 - 1/\ell)^{\ell-1} \ge 1/e$ that all other sampled points result in a smaller cost drop than c_{i+1}^j and thus the greedy rule decides that $c_{i+1} = c_{i+1}^j$. We claim this means that in the easy case, a sampled point from N_i means a constant probability of $|N_{i+1}| \le |N_i|/2$. To see this, let us order the points of N_i in their increasing distance from $\mu(K)$ as $n_1, n_2, \ldots, n_{|N_i|}$ (see Fig. 2). Importantly, if $c_{i+1}^j = n_t$, then N_{i+1} does not contain any of the points $n_{t+1}, n_{t+2}, \ldots, n_{|N_i|}$. Recall that all points of N_i have the same cost, up to constant factors. Hence, if we condition on $c_{i+1}^j \in N_i$, we know it is essentially a point of N_i selected uniformly at random. This means that with constant probability $c_{i+1}^j \in \{n_1, \ldots, n_{|N_i|/2}\}$ and hence $|N_{i+1}| \le |N_i|/2$, as we wanted.

To conclude, note that whenever we sample a point from N_i, we also hit K with probability $\varphi(K, C_i)/\varphi(N_i, C_i) \approx |K|/|N_i|$. But above discussion shows that a hit of N_i implies that $|N_{i+1}| \le |N_i|/2$ (with constant probability). Hence, the expected total number of hits of K of the easy case can be upper bounded by $\frac{|K|}{|X|} + \frac{|K|}{|X|/2} + \cdots + \frac{|K|}{2|K|} + \frac{|K|}{|K|} = O(1)$.

7

Hard case: Next, let us turn to the hard case. There, we at least know that with probability $1 - (1 - 1/\ell)^\ell \geq 1 - 1/e$, at least one candidate center c_{i+1}^j for $1 \leq j \leq \ell$ makes the cost drop by at least $\varphi(N_i, C_i)/2$. We hence get that $\varphi(X, C_i) - \mathrm{E}[\varphi(X, C_{i+1})] \geq (1 - 1/e)\varphi(N_i, C_i)/2 \geq \varphi(N_i, C_i)/4$.

Note that whenever it is the case that $\varphi(N_i, C_i) \leq \varphi(X, C_i)/k$, this implies that the probability we sample from N_i (and hence from K) is $\frac{\varphi(N_i, C_i)}{\varphi(X, C_i)} \leq 1/k$. Even if we sum up over all k steps, the total contribution in terms of number of samples from K is negligible. So, let us assume that $\varphi(N_i, C_i) \geq \varphi(X, C_i)/k$.

We will now consider the sequence of sampling steps of the algorithm until a step j when it selects a center from N_i. Until then, we have for each step $i + 1 \leq i' < j$ that $N_i = N_{i'}$, that is, the neighborhood of K is the same. Notice that the cost of all points in N_i also stays roughly the same during this time. This is because of our simplifying assumption that all points of N_i have distance at most $R_i/10$ from $\mu(K)$ and hence the cost of all points of N_i is between $(9R_i/10)^2$ and $(11R_i/10)^2$. This remains so even after sampling new centers with distance larger than R_i from K.

After $\varphi(X, C_i)/\varphi(N_i, C_i)$ steps, we expect to sample $O(\ell)$ many candidate centers from N_i. Meanwhile, the cost of X is expected to drop from $\varphi(X, C_i)$ to $3/4 \cdot \varphi(X, C_i)$ or smaller.

Recall that we assume that $\varphi(N_i, C_i) \geq \varphi(X, C_i)/k$. This means that after expected $O(\ell \log k)$ candidate centers sampled from N_i, we expect that $\varphi(X, C_{i'}) < \varphi(N_i, C_{i'})$ which is a contradiction. In other words, when we finally reach the step j when a center is picked from N_i, this center is some point out of $O(\ell \log k)$ candidate centers sampled from N_i so far between the steps i and j. Each of these candidates is essentially a uniformly random point of N_i (see Fig. 2).

We can now (as in the easy case) order the points of N_i as $n_1, \ldots, n_{|N_i|}$ in the order of increasing distance from $\mu(K)$. We sampled $O(\ell \log k)$ candidate centers from $\{n_1, \ldots, n_{|N_i|}\}$ essentially uniformly at random; hence with constant probability none of them hits the set of top $|N_i|/O(\ell \log k)$ points $\{n_{|N_i| - |N_i|/O(\ell \log k)}, \ldots, n_{|N_i|}\}$, hence with constant probability we have $|N_{i+1}| \leq |N_i|(1 - 1/O(\ell \log k))$.

That is, the size of $|N_i|$ is expected to drop by $1 - 1/O(\ell \log k)$ factor while $O(\ell \log k)$ candidate centers hit N_i, which corresponds to expected $\frac{|K|}{|N_i|} \cdot O(\ell \log k)$ hits of K. Put differently, after $\frac{|K|}{|N_i|} O(\ell^2 \log^2 k)$ hits of K we expect the size of the neighborhood N_i to halve.

Similarly to the easy case, this implies that the total number of hits of K can be upper bounded by $O(\ell^2 \log^2 k) \cdot \left(\frac{|K|}{|X|} + \frac{|K|}{|X|/2} + \cdots + \frac{|K|}{2|K|} + \frac{|K|}{|K|} \right) = O(\ell^2 \log^2 k)$. This finishes the hard case analysis and hence the whole proof.

The full proof of Lemma 2.2 is in Section 4. The only substantial difference from the above sketch is that as we do not have the simplifying assumption that all points of N_i are $R_i/10$-close to $\mu(K)$, we need to work with two sets N_i^{small}, N_i^{big} instead. We remark that one can replace the term $O(\ell^2 \log^2 k)$ by $O\left(\ell \log \frac{OPT(1)}{OPT(k)} \right)$ with a substantially easier proof, where $OPT(\widetilde{k})$ is the size of the optimal solution with $\widetilde{k}$ centers. We give a proof sketch in Appendix C.

2.4 Return of the guarantees

After proving Lemma 2.2, we are ready to prove the main result, Theorem 1.1. This is proven by adapting the k-means++ analysis of Arthur and Vassilvitskii [AV07]. As we have seen, their analysis gives $O(\log k)$ approximation guarantee. We pick up additional $O(\ell^2 \log^2 k)$ factor after using Lemma 2.2 to conclude that when a cluster K becomes covered, we expect its cost to be $5 \cdot \mathrm{E}[\mathrm{HIT}(K)] \cdot \varphi^*(K) = O(\ell^2 \log^2 k) \cdot \varphi^*(K)$. Finally, we need one more ℓ factor since the probability of a bad step can now be bounded only by $\frac{\ell \varphi(X_i^{\mathcal{U}}, C_i)}{\varphi(X, C_i)}$ instead of $\frac{\varphi(X_i^{\mathcal{U}}, C_i)}{\varphi(X, C_i)}$. This results in $O(\ell^3 \log^3 k)$ expected approximation guarantee.
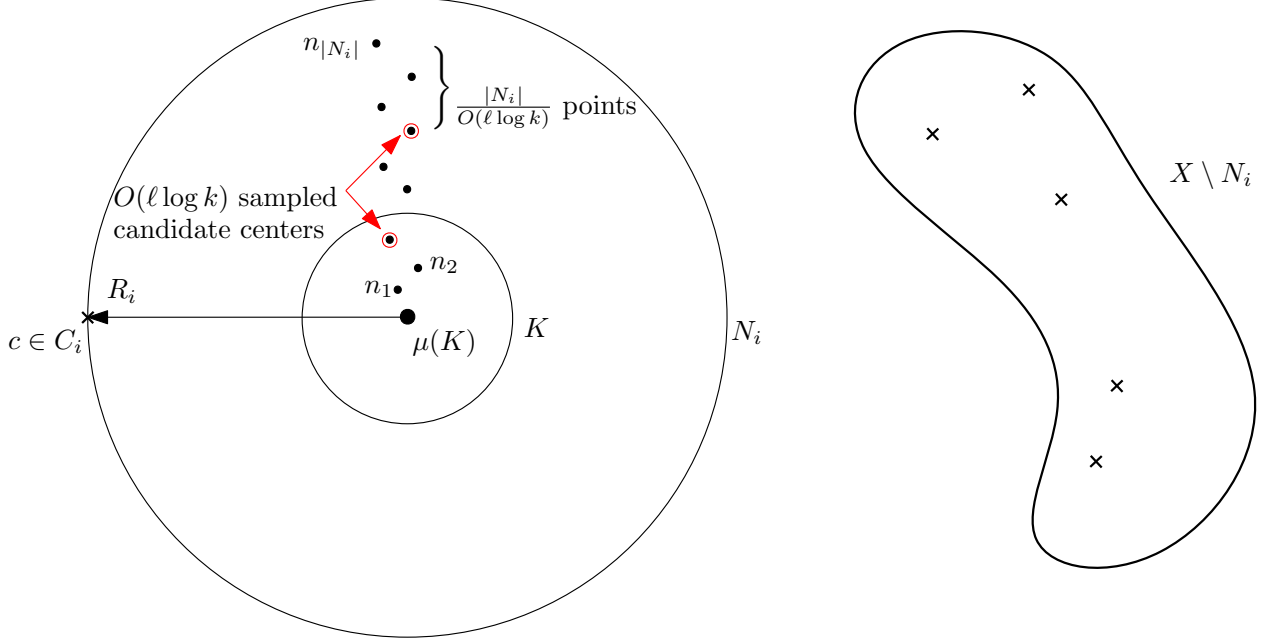
Figure 2: The figure shows an optimal cluster K and its neighborhood N_i. The points of N_i are sorted as $n_1, \ldots, n_{|N_i|}$ based on their distance from $\mu(K)$. Note that the picture is not to scale as our simplifying assumption says that even $n_{|N_i|}$ has distance at most $R_i/10$ from $\mu(K)$.

During the steps $i' \geq i$ until the step j with $N_j \neq N_i$, the relative cost $\varphi(N_i, C_{i'})/\varphi(X, C_{i'})$ of the neighborhood N_i increases from at least $1/k$ to at most 1, we expect to sample $\ell \log k$ candidate centers from N_i (highlighted by red color). One of these samples needs to be chosen as the new center in the step j that defines the new $N_j \neq N_i$. Since the red points are chosen essentially uniformly at random, we expect the top $|N_i|/(\ell \log k)$ points of N_i to be removed from N_{i+1}.

The above discussion may suggest that everything simply falls in place but there is one important subtlety that our analysis needs to deal with. In the k-means++ analysis, we paid the cost of $\varphi(X_i^{\mathcal{U}}, C_i)/u_i$ in every bad step. Recall that this was substantiated by the fact that the size of an average uncovered cluster is only expected to go down in the future steps, i.e., for k-means++ we can prove that $\mathrm{E}\left[\varphi(X_{i+1}^{\mathcal{U}}, C_{i+1})/u_{i+1}\right] \leq \varphi(X_i^{\mathcal{U}}, C_i)/u_i$. However, this bound is not necessarily true for $\ell > 1$. In fact, in the case of an adversarial rule, one can imagine the average size of a cluster increases substantially during the algorithm. Whether it is indeed so is an exciting open problem described in Appendix B.2. This is also the reason behind the mismatch in our upper and lower bounds of Theorems A.1 and B.1.

Fortunately, the fact that our rule is greedy and not an arbitrary one saves us again. To see this, first assume that instead of the greedy rule we use a different, idealized, rule that picks the candidate center c_{i+1}^j that minimizes the expression $\varphi(X_{i+1}^{\mathcal{U}}, C_i \cup \{c_{i+1}^j\})$ instead of $\varphi(X, C_i \cup \{c_{i+1}^j\})$ like the greedy rule. For such an idealized algorithm we have

$$\mathrm{E}\left[\varphi(X_{i+1}^{\mathcal{U}}, C_{i+1})/u_{i+1}\right] \leq \mathrm{E}\left[\varphi(X_{i+1}^{\mathcal{U}}, C_i \cup \{c_{i+1}^1\})/u_{i+1}\right] \leq \varphi(X_i^{\mathcal{U}}, C_i)/u_i.$$

The first inequality above is saying that our rule picks the best candidate c_{i+1}^j which is at least as good as the first candidate c_{i+1}^1. The second inequality is just the reasoning of the original k-means++ analysis. Therefore, for this idealized algorithm, the original analysis of k-means++ immediately generalizes.

Fortunately, our greedy rule is almost this idealized algorithm! The difference between the idealized minimization of $\varphi(X_{i+1}^{\mathcal{U}}, C_{i+1})$ and the actual greedy minimization of $\varphi(X, C_{i+1})$ just

creates two small mismatches: When the greedy rule considers a candidate center $c_{i+1}^j \in K$, it, in addition to the idealized rule, takes into account 1) the decrease in the cost of already covered clusters $\varphi(X_i^{\mathcal{C}}, C_i) - \varphi(X_i^{\mathcal{C}}, C_i \cup \{c_{i+1}^j\})$ and 2) the cost of the newly covered cluster $\varphi(K, C_i \cup \{c_{i+1}^j\})$. To give an example of the point (1), we can have a cluster K with $\varphi(K, C_i) \approx 0$ that is very close to covered clusters. When we sample $c_{i+1}^j \in K$ and we have $\varphi(X_i^{\mathcal{C}}, C_i \cup \{c_{i+1}^j\}) \ll \varphi(X_i^{\mathcal{C}}, C_i)$; for the greedy rule then taking c_{i+1}^j is an attractive option although taking it increases the average cost of the remaining uncovered clusters.

Fortunately, the two mismatches can be handled. After some calculations, it turns out that the first mismatch implies that we need to pay an additional cost of $\varphi(X_i^{\mathcal{C}}, C_i)/u_i$ per step in the analysis, but fortunately we already pay this term in the original analysis because of the possibility that the $i + 1$th step can be bad.

One can also show that the second mismatch implies that we need to pay additional factor of $\sum_{K \in \mathcal{K}_i^{\mathcal{U}}} \frac{\varphi(K, C_i)}{\varphi(X, C_i)} \varphi^*(K)$ where $\mathcal{K}_i^{\mathcal{U}}$ is the set of uncovered clusters. Fortunately, if we sum this expression over all steps, this is simply counting the number of hits to each optimal cluster K. So, in total, we need to pay additional term $\sum_{K \in \mathcal{K}_0^{\mathcal{U}}} \mathrm{E}[\mathrm{HIT}(K)] \varphi^*(K)$ in the approximation guarantee, but we are again already paying this term anyway because this is our upper bound on the cost of a cluster K once it becomes covered.

To conclude, the mismatch between the idealized algorithm and the actual greedy algorithm can be accounted for and the increase in the approximation factor is asymptotically dominated by terms we already have to pay in the analysis anyway for different reasons.

2.5 Matching lower bound

At this point, we have already a good understanding of where different terms in the approximation guarantee $O(\ell^3 \log^3 k)$ are coming from. This allows us to construct a point set where the above analysis is close to tight.

Point set definition: We describe the point set X next (see also Fig. 3) and then explain intuitively what happens when the greedy k-means++ is run on it. We will describe the lengths and weights here up to factors of size $O(\log(\ell \log k))$ that need to be added to make the construction work. Given a parameter k, there are $1 + \widetilde{k} = O(k^{1.2})$ points in X and we ask for a solution with $\widetilde{k}$ centers, hence exactly one point of X is going to be excluded. We set $t = \ell \log k$.

1. There is a point b for which we have $w(b) = \frac{1}{t^2}$.

2. A point c is at distance 1 from b. We have $w(c) = 1$. The points $\{b, c\}$ form one cluster of the optimal solution. While the optimal solution would take c as a center and would not take b, we will argue that greedy k-means++ takes b as a center with constant probability.

3. We have a set of points $N = \{n_1, n_2, \ldots, n_{t+1}\}$ and $M = \{m_1, \ldots, m_t\}$ defined as follows. We have $d(b, n_i) = k^i, d(n_i, m_i) = tk^i$ and $w(n_i) = w(m_i) = \frac{1}{t}$. An exception is the point n_{t+1} that gets very large weight so that it is selected as a center in the first two steps of the algorithm.

4. We have $E = \{e_0, \bigcup_{e \in E_1} e, \ldots, \bigcup_{e \in E_t} e\}$ where $E_i = \{e_{i,1}, \ldots, e_{i,\sqrt{k}}\}$. Each point in $E \setminus \{e_0\}$ has distance 1 from the point e_0 which is very far from $\{b, c\} \cup N \cup M \cup A$. The point e_0 has very large weight so that it is selected as a center in the first two steps of the algorithm. Each $e_{i,j} \in E_i$ has the same weight $w_i \approx k^{2i+2}$. We postpone the exact definition of w_i as it needs to be quite precise.

Figure 3: This figure shows the point set X used for the lower bound (up to small changes by poly $\log(\ell \log k)$ factors). The point weights are orange and heavier points are represented by larger disks.

In this intuitive explanation, the metric is not Euclidean but it is a tree (or a bit more precisely forest) metric induced by the blue edges with the distances given by blue numbers. The image is not to scale, for example, the distance from b to n_t is in fact much larger than the distance from b to m_2.

In the first two steps, the points n_{t+1} and e_0 are taken. During the ith phase of the first epoch we mostly take just points of $E_i = \{e_{i,1}, \ldots, e_{i,\sqrt{k}}\}$ that serve as a "clock". This clock is ticking for enough time so that the algorithm samples n_i as a candidate. Because of m_i, n_i is then selected by the greedy rule. This drastically reduces the costs of the not-yet-taken points of $X \setminus E$ so that nothing interesting happens until all points of E_i are taken (the clock for this phase stops ticking) and then we go to the next phase. The aim of these phases is that in each one of them we have a small probability of $1/t$ of sampling b and taking it as the center. As there are t phases, we get constant probability of taking it.

In the second epoch, we simply show that the greedy k-means++ samples and then takes c with constant probability, which increases the approximation factor by additional $\ell \log k$.

5. We have $A = \{a_1, \ldots, a_{k^{1.2}}\}$ at distance k from c. The weight of each a_i is $\frac{\ell \log k}{k^2}$ so their total weight is $\frac{\ell \log k}{k^{0.8}}$.

The optimal solution on X takes all the points except b as centers and hence it pays $1/(\ell^2 \log^2 k)$.

Here is an intuitive explanation how greedy k-means++ runs on X. There are two epochs. In the first epoch, the set A is too small to be discovered by the algorithm, so only $\{b, c\} \cup N \cup M \cup E$ is relevant. Our aim is to show that with constant probability we reach a situation where only the points in $A \cup \{c\}$ are not taken; when this happens we say that the second epoch starts.

Note that $K = \{b, c\}$ is a cluster in the optimal solution. Hence, we are proving that in the first epoch the cost of K under the greedy k-means++ is approximated only by factor $\Omega(\ell^2 \log^2 k)$, matching the bound in the proof of Lemma 2.2. Moreover, for each i the sets $\{n_1, m_1, \ldots, n_i, m_i\}$ are playing the role of the neighborhood N_i of K in the analysis from Section 2.3, while the set E plays the role of $X \setminus K$.

First epoch : Here is what is going to happen in the first epoch. We can split it into t phases

where at the beginning of the ith phase all the points in $N_{\geq i+1} \cup M_{\geq i+2} \cup E_{\geq i+1}$ [5] are already selected as centers. During the i-th phase, we do not sample points from $A \cup E_{<i}$ as they have too small a probability of being sampled. Mostly, we sample just points of E_i since each point there has cost about k^{2i+2}. These points serve as a kind of clock. During the time we are sampling mostly points of E_i, we also have a small chance of sampling points of $\{b,c\} \cup N_{\leq i} \cup M_{\leq i+1}$. Since we start with $|E_i| \geq \sqrt{k}$, before we add all of E_i to the set of centers, we expect to sample the point c about $\ell \log k$ times. Similarly, each point in $N_{\leq i} \cup M_{\leq i+1}$ is expected to be sampled constantly many times. The point b has only probability of around $(\ell \log k)/t^2 = 1/t$ of being sampled.

Now we can fine-tune the weight of points in E_i so that the drop in the cost if we take $e \in E_i$ is larger than the drop resulting from taking c but smaller than the drop of b (taking b results in a larger drop than c, since c is further from $N_{\leq i} \cup M_{\leq i}$ than b). This means that the drop of e is smaller than the drop of the points $\{b, n_i, m_{i+1}\}$, but it is larger than the drop of all other points. In fact, the reason why we always have a pair $\{n_i, m_i\}$ is to make the cost drop of n_i large: as n_i lies roughly in the center of mass of c and m_i, it is a very attractive point to take from the perspective of the greedy rule. So, during the ith phase we are essentially just waiting until we sample n_i. When we encounter it, we add it to the set of centers which decreases the cost of $\{b,c\} \cup N_{<i} \cup M_{\leq i}$ dramatically so that in the rest of the phase only the points of E_i are sampled. Also, the point m_{i+1} that is leftover from the previous phase is at some point sampled and selected as a center.

This process is running for t phases and in each phase, we have probability of $1/t$ of sampling b. After b is sampled, all other points except for $A \cup \{c\}$ are sampled in the following steps; the weight of all of them is much larger than the weight of points of $A \cup \{c\}$. When this is done, the first epoch is finished.

Second epoch: While the first epoch corresponds to the bound that we lose in Lemma 2.2 for not approximating well clusters that get covered, the second epoch corresponds to the rest of the analysis in which we lose additional $\ell \log k$ factor because of the fact that some steps are bad. In our case, a bad step means that we select a point c from the optimal cluster $K = \{b, c\}$ that we already covered in the first epoch by selecting b.

The second epoch begins when all points except of $A \cup \{c\}$ are already taken as centers. Note that in the following $k^{1.2} - k^{1.1}$ steps we have a constant probability that we sample c as a candidate center. In that case, the greedy heuristic decides to pick c over the other candidates from A. This is because the cost drop induced by any $a \in A$ is simply $w(a) \cdot d(a,b)^2 \approx \ell \log k$, whereas selecting c makes the cost of each $a \in A$ drop by roughly $w(a) \cdot \big((k+1)^2 - k^2\big) \approx \frac{\ell \log k}{k^2} \cdot k = \frac{\ell \log k}{k}$. However, there are at least $k^{1.1}$ points in A, hence the drop of c is larger than the drop of any $a \in A$.

Putting it together: Putting the two epochs together, we get a constant probability that the greedy k-means++ algorithm first fails by covering K using b, instead of c, and then it fails again by taking c although K is already covered. This means that one point of A is not covered in the end and we need to pay $\ell \log k$ for it, whereas the optimum opts not to take b and hence pays only $1/(\ell^2 \log^2 k)$.

One problem with the above analysis is that at each epoch we have a constant probability of not sampling n_i before all points of E_i are added to the set of centers. In that case, our analysis fails since it is probable that the algorithm will soon afterward sample c and take it as a center. However, adjusting weights by $\mathrm{poly}(\log t) = \mathrm{poly}\log(\ell \log k)$ factors makes the failure probability of one phase smaller than $1/(2t)$ so that we can union bound over them.

---

[5] The notation $N_{\geq i+1}$ means $\{n_{i+1}, \ldots, n_t\}$.

3   Preliminaries

We use X for the input point set. In case X is weighted as discussed below in the lower bound section, every point $x \in X$ comes with nonnegative weight $w(x)$, in the unweighted case we have $w(x) = 1$. Whenever we talk about an optimal cluster K, we tacitly assume a fixed optimal solution C^* and $K \subseteq X$ is a set of points defined by some $c \in C^*$ as $K = \{x \in X : c = \operatorname{argmin}_{c' \in C^*} \varphi(x, c')\}$.

We use $d(x, y)$ to denote the distance between two points x, y rather then $||x - y||$, since all our upper bounds generalize to general metric space (the only difference being that Lemmas 3.2 and 3.3 require larger constant factors in that case).

For $x \in X$ we use $\varphi(x, C) = w(x) \cdot \min_{c \in C} d(x, c)^2$ and for $K \subseteq X$ we define $\varphi(K, C) = \sum_{x \in K} \varphi(x, C)$. For a cluster K we write $\varphi^*(K) = \varphi(K\mu(K))$ where $\mu(K)$ is the center of mass of K, i.e., $\mu(K) = (\sum_{x \in K} x)/w(K)$ for $w(K) = \sum_{x \in K} w(x)$. That is, $\varphi^*(K)$ is the smallest cost of K achievable if we have just one center.

When we write $\mathrm{E}_{\geq i+1}[X]$ in relation to Algorithms 1, 2 and 3, we tacitly assume that the randomness of the first i steps is fixed and the expectation is over the randomness in the rest of the algorithm. Similarly, $\mathrm{E}_{i+1}[X]$ is an expectation over the randomness in the step $i + 1$.

Standard lemmas from [AV07]: We will need some lemmas from [AV07].

First, note that for the squared distance we have the approximate triangle inequality

$$d(x, z)^2 \leq 2(d(x, y)^2 + d(y, z)^2) \tag{1}$$

The following lemmas can be found in [AV07].

Lemma 3.1 (Lemma 2.1 in [AV07]). *For any K, c we have $\varphi(K, c) = \varphi^*(K) + |K|c^2$.*

Lemma 3.2 (Lemma 3.1 in [AV07]). *If c is a uniformly randomly selected point of K, we have $E[\varphi(K, c)] = 2\varphi^*(K)$.*

Lemma 3.3 (Lemma 3.2 in [AV07], Lemma 4.2 in [MRS20]). *Fix two point sets K, C and sample a random points $c \in K$ with probability proportional to $\varphi(c, C)$. Then,*

$$E[\varphi(K, C \cup \{c\})] \leq 5\varphi^*(K).$$

We note that the original version of the lemma from [AV07] had the constant being 8, this was improved to 5 in the work of [MRS20].

Lemmas for lower bounds: For lower bounds, it will be easier to work with the more general weighted version of the k-means problem. Any lower bound for the weighted version can be lifted to an unweighted one by multiplying all weights by a large number and rounding them to the closest integer.

Even more generally, it will be more convenient to prove our lower bounds for the generalized k-means problem where the input contains not only a weighted point set X and k, but also a set of prescribed centers C_0. We next use (X, k, C_0) to denote input to such generalized problem. We will use the following fact.

Lemma 3.4. *Suppose that any of the algorithms Algorithms 1, 2 and 3 returns at least α-approximate solution on some k-means instance (X, k, C_0) with constant probability. Then there exists an instance $(X', k + |C_0|, \emptyset)$ where the algorithm is still at least α-approximate with constant probability.*

Proof Sketch. We simply define X' to be $X \cup C_0$, where we make the weight of each point in C_0 substantially larger than the total weight of X so that the first $|C_0|$ steps only points of C_0 can be sampled, with high probability. $\square$

Above construction can, in general, substantially increase the weights. However, we only use it in case $|C_0| \leq 2$ where this is not the case.

Another useful fact is that if we restrict our solution to k-means to points of X, we lose only a 2-factor in approximation. This lemma follows directly from Lemma 3.2.

Lemma 3.5. *Whenever there is some solution C_k to an instance of k-means, there is also a solution $C'_k \subseteq X$ such that $\varphi(X, C'_k) \leq 2\varphi(X, C_k)$.*

Finally, our lower bounds would be easier if they were proven in general metric spaces. It is simple, though a bit technical, to make them work in Euclidean spaces. To this end, we need the following result about embedding a star metric to the Euclidean space.

Fact 3.6. *There is a way to arrange d vectors $v_1, \ldots, v_d$ in Euclidean space in such that for any two v_i, v_j, we have $\langle v_i, v_j \rangle = -1/(d-1)$. In particular, we get this property by arranging $v_1, \ldots, v_d$ as vertices of a $d-1$ dimensional simplex.*

Few more results: All logarithms in this paper are natural. We will use the following facts.

Fact 3.7. *For any $x > 0$ we have $\log(x) \geq 1 - 1/x$.*

Fact 3.8. *For any $x \in [-1, 0]$ we have $\left(1 + \frac{x}{2}\right) \geq e^x$.*

We also need the following lemma that says that if we learn about a random variable that it is larger than some other independently sampled variables it can only increase its expectation.

Lemma 3.9. *Let $X, Y_1, \ldots, Y_t$ be independent random variables. We have*

$$E[X | X = \min(X, Y_1, \ldots, Y_t)] \leq E[X].$$

Proof. We assume all variables are discrete. Let us consider any $(y_1, \ldots, y_t) \in \mathbb{R}^t$, and fix $Y_i = y_i$ for all $1 \leq i \leq t$. We have that the conjunction of the event $\forall i : Y_i = y_i$ together with the event $X = \min(X, Y_1, \ldots, Y_t)$ is equivalent to the conjunction of the event $\forall i : Y_i = y_i$ with $X \leq \min(y_1, \ldots, y_t)$. Hence, we can write

$$E[X | X = \min(X, Y_1, \ldots, Y_t)] = \sum_{(y_1, \ldots, y_t) \in \mathbb{R}^t} P(\forall i : Y_i = y_i) \cdot E\left[X | X = \min(X, Y_1, \ldots, Y_t) \wedge \forall i : Y_i = y_i\right]$$

$$= \sum_{(y_1, \ldots, y_t) \in \mathbb{R}^t} P(\forall i : Y_i = y_i) \cdot E\left[X | X \leq \min(y_1, \ldots, y_t) \wedge \forall i : Y_i = y_i\right]$$

$$= \sum_{(y_1, \ldots, y_t) \in \mathbb{R}^t} P(\forall i : Y_i = y_i) \cdot E\left[X | X \leq \min(y_1, \ldots, y_t)\right]$$

However, for any $(y_1, \ldots, y_t) \in \mathbb{R}^t$ we have $E\left[X | X \leq \min(y_1, \ldots, y_t)\right] \leq E[X]$ and the lemma follows. $\square$

4 Bounds on cluster hits by greedy k-means++

In this section we give a formal proof of Lemma 2.2. We start by giving preparatory definitions and the statement of Lemma 4.7 – inductive variant of Lemma 2.2 – in Section 4.1. The lemma is then proved in Section 4.2.

4.1 Preparatory definitions and results

We start by formally defining covered and solved clusters.

Definition 4.1 (Covered Cluster). *Consider some optimal cluster K. We refer to K as covered with respect to a set of centers C if $K \cap C \neq \emptyset$.*

Definition 4.2 (Solved Cluster). *Consider some optimal cluster K. We refer to K as solved with respect to point set C if*

$$\varphi(K, C) \leq 10^5 \varphi^*(K).$$

We also say that C is covered/solved in the $i + 1$th step of the algorithm if it is covered/solved with respect to C_i.

Next, we formally define the object of interest, that is, the number of points we are going to sample from K during Algorithm 2.

Definition 4.3 (Number of points we will sample). *Let $HIT_{i+1}^j(K)$ be an indicator variable for the conjunction of the following three events:*

  1. $c_{i+1}^j \in K$

  2. K is not covered with respect to C_i

  3. K is not solved with respect to C_i

We further define $HIT_i(K) := \sum_{j=1}^{\ell} HIT_i^j(K)$, $HIT_{\geq i}(K) = \sum_{\iota=i}^{k} HIT_\iota(K)$ and $HIT(K) = HIT_{\geq 1}(K)$.

Recall that Lemma 2.2 asks us to prove that $\mathrm{HIT}(K) = O(\ell^2 \log^2 k)$.

We will need a few more definitions to state our main technical result Lemma 4.7 which is an inductive version of Lemma 2.2. We start with the definition of the parameter R_i. This parameter is, up to constant factors, equal to the distance $d(\mu(K), C_i)$ of the center of mass of K with the closest center of C_i. However, it may be that $R_i = R_{i+1}$ although $d(\mu(K), C_i) = d(\mu(K), C_{i+1})$. This is because whenever R_i changes, we want it to change by a factor of 10 for a reason explained later.

We also define the index i_0 as the smallest index of a step where the size of K becomes "non-negligible". Before step i_0, we have only a negligible probability of sampling a point from K. We note that from now on, we consider the cluster K fixed, so that we can talk about R_i instead of $R_i(K)$, etc.

Definition 4.4 (Parameters i_0 and R_i). *Fix an optimal cluster K. Let $i_0 = \min\{i \in \{1, 2, \ldots, k\} : \varphi(K, C_i) \geq \varphi(X, C_i)/k\}$. For every $i \in \{i_0, i_0 + 1, \ldots, k\}$, we define a parameter R_i with*

$$R_{i_0} = d(\mu(K), C_{i_0})$$

and for $i > i_0$, we define

$$R_i = \begin{cases} R_{i-1} & d(\mu(K), C_i) > R_{i-1}/10 \\ d(\mu(K), C_i) & otherwise. \end{cases}$$

Note that R_i satisfies

$$d(\mu(K), C_i) \leq R_i \leq 10d(\mu(K), C_i). \tag{2}$$

15

That is, R_i is roughly equal to the distance of the cluster center to the set C_i. Next, we always implicitly assume that $i \geq i_0$, i.e., R_i is well defined.

Based on R_i we also define

$$N_i^{small} = B(\mu(K), R_i/100) \tag{3}$$

and

$$N_i^{big} = B(\mu(K), R_i/10). \tag{4}$$

These definitions correspond to the neighborhood N_i from the intuition section in Section 2.3. We need two different neighborhoods due to technical difficulties like the fact that only for $x \in N_i^{small}$ we have $d(x, C) = \Omega(R_i)$ (in the proof sketch of Section 2.3 we worked under a simplifying assumption that all $x \in N_i^{big}$ are also in N_i^{small}). The reason for the slightly weird definition of R_i is that we want that whenever $R_{i+1} < R_i$, then $N_{i+1}^{big} \subseteq N_i^{small}$.

In the following claim, we summarize several basic properties of the cluster K and its neighborhoods N_i^{small} and N_i^{big}. These claims substantiate some simple intuition about these sets like that all points in N_i^{small} have essentially the same cost. A careful reader will note that we do not try to optimize the constants (and we warn that it will get worse).

Claim 4.5. *Assume K is not solved with respect to C_i for $i \geq 1$. Then, we have:*

1. *For each point $x \in N_i^{small}$, we have*

$$R_i^2/200 \leq \varphi(x, C_i) \leq 3R_i^2.$$

and for each point $x \in N_i^{big}$ we have

$$0 \leq \varphi(x, C_i) \leq 3R_i^2.$$

2. *We have*

$$|N_i^{small}| \cdot R_i^2/200 \leq \varphi(N_i^{small}, C_i) \leq 3|N_i^{small}| \cdot R_i^2$$

and

$$\varphi(N_i^{big}, C_i) \leq 3|N_i^{big}| \cdot R_i^2.$$

3. *We have*

$$|K|R_i^2/400 \leq \varphi(K, C_i) \leq 2|K|R_i^2.$$

4. *At least $|K|/2$ points of K are in N_i^{small}. Also, $\varphi(K \cap N_i^{small}, C_i) \geq \varphi(K, C_i)/800$.*

Proof. Let $c^* = \operatorname{argmin}_{c \in C_i} d(\mu(K), c)$. Recall that Eq. (2) implies that

$$d(\mu(K), c) \leq R_i \leq 10d(\mu(K), c). \tag{5}$$

1. Consider any $x \in N_i^{big}$. We have

$$\varphi(x, C_i) \leq d(x, c^*)^2 \leq 2(d(x, \mu(K))^2 + d(\mu(K), c^*)^2) \qquad \text{Eq. (1)}$$

$$\leq 2\left(\left(\frac{R_i}{10}\right)^2 + R_i^2\right) \qquad \text{Eq. (5)}$$

$$\leq 3R_i^2$$

On the other hand, for any $x \in N_i^{small}$, we have

$$\varphi(x, C_i) \geq (d(\mu(K), c^*) - d(\mu(K), x))^2$$

$$\geq (R_i/10 - R_i/100)^2$$

$$\geq R_i^2/200.$$

16

2. This follows by applying bullet point (1) to every point in N_i^{small} and N_i^{big}, respectively.

3. Recall that by Lemma 3.1, we have

$$\varphi(K, C_i) \leq \varphi(K, c^*) = |K| \cdot d(\mu(K), c^*)^2 + \varphi^*(K) \leq |K| R_i^2 + \varphi^*(K). \tag{6}$$

On the other hand,

$$\varphi(K, C_i) \geq 10^5 \varphi^*(K)$$

by Definition 4.2. Hence,

$$\varphi^*(K) \leq \frac{|K| R_i^2}{10^5 - 1}. \tag{7}$$

If we plug Eq. (7) back to Eq. (6), we get

$$\varphi(K, C_i) \leq |K| R_i^2 + \frac{|K| R_i^2}{10^5 - 1} \leq 2|K| R_i^2.$$

We postpone the proof of the other inequality to the end of the proof.

4. Define d by $\varphi^*(K) = |K| d^2$, that is, d is the average squared distance of a point of K to $\mu(K)$. Note that Eq. (7) implies that $d^2 \leq R_i^2/(10^5 - 1)$. By Markov's inequality, at most $|K|/2$ points can have a cost of at least $2d^2$, which is at most $2R_i^2/(10^5 - 1) \leq (R_i/100)^2$. Hence, at least $|K|/2$ points of K need to be in N_i^{small}.

Moreover, using the above result and bullet point (2), we get

$$\varphi(K \cap N_i^{small}, C_i) \geq \frac{|K|}{2} \cdot \frac{R_i^2}{200}$$

and using the part of bullet point (3) that we have already proven, we have

$$2|K| R_i^2 \geq \varphi(K, C_i).$$

Combining the two bounds, we get

$$\varphi(K \cap N_i^{small}, C_i) \geq \frac{|K| R_i^2}{400} \geq \frac{\varphi(K, C_i)}{800}.$$

5. Finally, using the fact that at least $|K|/2$ points of K are in N_i^{small} by bullet point (4) and that each point $x \in N_i^{small}$ satisfies $\varphi(x, C_i) \geq R_i^2/200$ by bullet point (1) we infer that $\varphi(K, C_i) \geq |K|/2 \cdot R_i^2/200 = |K| R_i^2/400$ and the proof of bullet point (3) is finished.

□

The reason why we deal with the steps before i_0 differently is that for all $i \geq i_0$ we know that $|N_i^{big}|$ is at most $O(k)$ times larger then $|K|$.

Claim 4.6. *Let i_0 as defined in Definition 4.4. For all $i \geq i_0$, we have*

$$|N_i^{big}| \leq 4k|K|.$$

Proof. It suffices to prove that $|N_{i_0}^{big}| \le 4k|K|$, since Definition 4.4 implies that $|N_{i+1}^{big}| \le |N_i^{big}|$ for any i.

Recall that $R_{i_0} := d(\mu(K), C_{i_0})$. Consider any $x \in N_{i_0}^{big}$. We have

$$\varphi(x, C_{i_0}) \ge (d(C_{i_0}, \mu(K)) - d(\mu(K), x))^2 \ge (R_{i_0} - R_{i_0}/10)^2 \ge R_{i_0}^2/2. \tag{8}$$

This implies that

$$\varphi(X, C_{i_0}) \ge \varphi(N_{i_0}^{big}, C_{i_0}) \ge |N_{i_0}^{big}| \cdot R_{i_0}^2/2. \tag{9}$$

One the other hand, using bullet point (3) of Claim 4.5 we get that

$$2|K|R_{i_0}^2 \ge \varphi(K, C_{i_0}). \tag{10}$$

By definition of i_0 we know that $\varphi(K, C_{i_0}) \ge \varphi(X, C_{i_0})/k$. Putting this together with Eqs. (9) and (10), we get

$$2|K|R_{i_0}^2 \ge |N_{i_0}^{big}| \cdot R_{i_0}^2/(2k)$$

or

$$|N_{i_0}^{big}| \le 4k|K|,$$

as needed. □

4.2 Inductive version of the main result

Finally, we are ready to state the technical, inductive version of Lemma 2.2. The lemma is a potential argument: we cook up three potentials such that their sum in the step i is an upper bound on how many candidate centers are expected to be sampled from K. The advantage of such an argument is that once the potentials are written down, checking the correctness of the argument reduces to algebra. The disadvantage is that it is hard to get an intuition about the high-level picture (compare with the original analysis of k-means++ and its rewording of Dasgupta [Das19]). Hence, we first invite the reader to read Section 2.4 where we try to convey the high-level intuition about the argument. This section is primarily optimized for making it easy to check the correctness of the proofs.

In Lemma 4.7, we track the number of hits to K using three potentials.

1. The first potential π_i is "dropping" a potential of $O(\ell/k)$ uniformly in every step. This allows us to argue about some edges cases like when $\varphi(K, C_i) < \varphi(X, C_i)/k$. In these cases we have probability of at most ℓ/k of hitting K and the drop in π_i can pay for that.

2. Whenever the value of R_i changes, we "refill" the value of potential σ_i to its maximum value of $O\left(\ell \log k \cdot \frac{|K|}{|N_i^{small}|}\right)$. The purpose of ρ_i is to pay for this refill of σ_i. Intuitively, we hope that whenever R_i drops, the size of $|N_i^{big}|$ drops by at least $(1 - O(1/\ell \log k))$ multiplicative factor (see the intuition in Section 2.3). In that case, the drop in ρ_i is proportional to exactly $O(\frac{\ell^2 \log^2 k}{\ell \log k} \cdot \frac{|K|}{|N_i^{small}|})$, so it can pay for the refill of σ_i.

3. The third potential σ_i is designed to pay for the possibility of hitting K until we redefine R_i. Intuitively, if we are in the hard case when hitting K does not result in covering it, we are expecting the cost of our solution to drop. Accordingly, drop in the overall cost $\varphi(X, C_i)$ results in the drop in σ_i and this drop is paying for the possibility of hitting K.

18

Lemma 4.7. *Fix an optimal cluster K. Assume we have already sampled the first i points where $i \in \{1, 2, \ldots, k\}$.*

Then, we have

$$E_{\geq i+1}[HIT_{\geq i+1}(K)] \leq \pi_i + \rho_i + \sigma_i,$$

where

$$\pi_i = \left(1 - \frac{i}{k}\right) \cdot \ell + 10^{50} \ell \log k,$$

$$\rho_i = 10^{50} \ell^2 \log^2(k) \cdot \left(4 - \frac{|K|}{|N_i^{small}|} - \frac{|K|}{|N_i^{big}|}\right)$$

and

$$\sigma_i = 10^{25} \ell \cdot \log \min\left(10^5 k, \frac{400\varphi(X, C_i)}{|K|R_i^2}\right) \cdot \frac{|K|}{|N_i^{small}|}$$

This is only when K is neither solved nor covered with respect to C_i. Otherwise, we define $\pi_i = \rho_i = \sigma_i = 0$.

Before proving Lemma 4.7, we briefly verify that it implies Lemma 2.2:

Proof of Lemma 2.2. First, note that in the first step we sample at most ℓ points from K, that is, $HIT_1 \leq \ell$.

Next, we consider running Algorithm 2 until the first time it happens that

$$\varphi(K, C_i) > \varphi(X, C_i)/k. \tag{11}$$

Fix some i such that Eq. (11) does not hold and it did not hold for any $\iota \in \{1, 2, \ldots, i-1\}$. Then, the expected number of points we sample from K in the $(i+1)$-th step is at most

$$\text{E}[HIT_{i+1}] \leq \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} \leq \frac{\ell}{k}. \tag{12}$$

In fact, the first inequality is an equality, whenever the cluster K is not solved or covered. The second inequality uses our assumption that $\varphi(K, C_i) \leq \varphi(X, C_i)/k$.

Next, consider the first i such that Eq. (11) holds. Then, we apply Lemma 4.7. This lemma states that $\text{E}[HIT_{\geq i+1}]$ can be upper bounded by a sum $\pi_i + \rho_i + \sigma_i$. From the definition of these quantities in Lemma 4.7 we immediately see that

$$\pi_i = O(\ell \log k),$$

$$\rho_i = O(\ell^2 \log^2 k)$$

and

$$\sigma_i = O(\ell \log k).$$

Hence, we get that

$$\text{E}[HIT_{\geq 1}] \leq \ell + k \cdot \frac{\ell}{k} + O(\ell^2 \log^2 k) = O(\ell^2 \log^2 k).$$

$\square$

The rest of this section is dedicated to the proof of Lemma 4.7.

We prove the statement by (reverse) induction. First, consider the base case $i = k$. Note that for the base case it suffices to show that $\pi_i, \rho_i, \sigma_i \geq 0$. If K is covered or solved with respect to C_i, then this directly follows from the definition. Next, assume that K is neither solved nor covered with respect to C_i. Clearly $\pi_{k+1} \geq 0$. Using bullet point (4) of Claim 4.5, we conclude that $|N_k^{big}| \geq |N_k^{small}| \geq |K|/2$ which implies $\rho_{k+1} \geq 0$. Using bullet point (3) of Claim 4.5, we conclude that $\varphi(X, C_k) \geq \varphi(K, C_k) \geq \frac{|K|R_k^2}{400}$ which implies $\sigma_{k+1} \geq 0$.

Next, we consider $i < k$. Note that the statement trivially holds if K is solved or covered with respect to C_i. Hence, from now on we consider the case that K is neither solved nor covered with respect to C_i.

Note that we have

$$\mathrm{E}_{\geq i+1}[\mathrm{HIT}_{\geq i+1}(K)] = \mathrm{E}_{i+1}[\sum_{j=1}^{\ell} \mathrm{HIT}_{i+1}^j] + \mathrm{E}_{i+1}\left[\mathrm{E}_{\geq i+2}[\mathrm{HIT}_{\geq i+2}(K)]\right] \tag{13}$$

By definition of HIT and the fact that K is both uncovered and unsolved, we have $\mathrm{E}_{i+1}[\sum_{j=1}^{\ell} \mathrm{HIT}_{i+1}^j] = \frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)}$ and we can use induction to bound

$$\mathrm{E}_{i+1}\left[\mathrm{E}_{\geq i+2}[\mathrm{HIT}_{\geq i+2}(K)]\right] \leq \mathrm{E}_{i+1}[\pi_{i+1} + \rho_{i+1} + \sigma_{i+1}]. \tag{14}$$

Plugging back to Eq. (13), we get

$$\mathrm{E}_{\geq i+1}[\mathrm{HIT}_{\geq i+1}(K)] \leq \frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)} + \mathrm{E}_{i+1}\left[\pi_{i+1} + \rho_{i+1} + \sigma_{i+1}\right]. \tag{15}$$

Note that the claim we want to prove is

$$\mathrm{E}_{\geq i+1}[\mathrm{HIT}_{\geq i+1}] \leq \pi_i + \rho_i + \sigma_i, \tag{16}$$

so it suffices if we prove that

$$\frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)} + \mathrm{E}_{i+1}\left[\pi_{i+1} + \rho_{i+1} + \sigma_{i+1}\right] \leq \pi_i + \rho_i + \sigma_i.$$

After rearranging, we get

$$(\pi_i - \mathrm{E}_{i+1}[\pi_{i+1}]) + (\rho_i - \mathrm{E}_{i+1}[\rho_{i+1}]) + (\sigma_i - \mathrm{E}_{i+1}[\sigma_{i+1}]) \geq \frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)}. \tag{17}$$

The potential σ is the only one of π, σ, ρ that is not necessarily monotone in i. Hence, to better understand the term $(\sigma_i - \mathrm{E}_{i+1}[\sigma_{i+1}])$, given the sampled point c_{i+1}, we define $\overline{\sigma_{i+1}}$ as

$$\overline{\sigma_{i+1}} = 10^{25}\ell \log \min\left(10^5 k, \frac{400\varphi(X,C_{i+1})}{|K|R_i^2}\right)\frac{|K|}{|N_i^{small}|}. \tag{18}$$

That is, in $\overline{\sigma_{i+1}}$ we already change the cost $\varphi(X, C_i)$ to $\varphi(X, C_{i+1})$ but we do not replace R_i by R_{i+1} and N_i^{small} by N_{i+1}^{small} yet. We can rewrite Eq. (17) and get

$$(\pi_i - \mathrm{E}_{i+1}[\pi_{i+1}]) + (\rho_i - \mathrm{E}_{i+1}[\rho_{i+1}]) + (\sigma_i - \mathrm{E}_{i+1}[\overline{\sigma_{i+1}}]) + \mathrm{E}_{i+1}[\overline{\sigma_{i+1}} - \sigma_{i+1}] \geq \frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)}. \tag{19}$$

In the rest of the proof, we will simply need to show that Eq. (19) is satisfied. We will need to consider several cases and in each one of them, we will have to lower bound the terms on the left-hand side of Eq. (19). Formally, the proof follows from Claims 4.11 and 4.16 to 4.18 that cover all possible cases that can occur.

4.3 Basic properties of potentials

Here we collect some basic claims about the potentials π, ρ and σ. We start with π. Note that we always have

$$\pi_i - \mathrm{E}_{i+1}[\pi_{i+1}] \geq \frac{\ell}{k} \tag{20}$$

by definition of π_i. Note that this is just an inequality. When K becomes solved or covered, we have

$$\pi_i - \pi_{i+1} \geq 10^{50} \ell \log k. \tag{21}$$

We continue with ρ whose changes we handle through the following claim. The main message of the claim is that whenever we have $R_i \neq R_{i+1}$ and, even more, $|N_{i+1}^{big}| \leq (1 - O(1/\ell \log k))|N_i^{big}|$, we have $\rho_i - \rho_{i+1}$ as large as the maximum size of σ_i.

Claim 4.8. *We have*

$$\rho_i - \rho_{i+1} \geq 10^{50} \ell^2 \log^2 k |K| \left(\frac{1}{|N_{i+1}^{small}|} - \frac{1}{|N_i^{small}|} + \frac{1}{|N_{i+1}^{big}|} - \frac{1}{|N_i^{big}|} \right) \tag{22}$$

In particular, we always have

$$\rho_i - \rho_{i+1} \geq 0.$$

Moreover, if we assume that

$$|N_{i+1}^{big}| \leq (1 - 1/(10^{20}\ell \log k))|N_i^{big}|,$$

then we have:

$$\rho_i - \rho_{i+1} \geq 10^{30}\ell \log k \frac{|K|}{|N_{i+1}^{small}|}$$

Proof. The first inequality follows from the definition of ρ.

Next, assume that

$$|N_{i+1}^{big}| \leq (1 - 1/(10^{20}\ell \log k))|N_i^{big}|. \tag{23}$$

We have

$$\left(\frac{1}{|N_{i+1}^{small}|} - \frac{1}{|N_i^{small}|} + \frac{1}{|N_{i+1}^{big}|} - \frac{1}{|N_i^{big}|} \right)$$

$$\geq \left(\frac{1}{|N_{i+1}^{small}|} - \frac{1}{|N_i^{big}|} \right) \qquad N_{i+1}^{big} \subseteq N_i^{small} \text{ by definition of } R_i$$

$$= \frac{|N_i^{big}| - |N_{i+1}^{small}|}{|N_i^{big}||N_{i+1}^{small}|}$$

$$\geq \frac{\frac{1}{10^{20}\ell \log k} \cdot |N_i^{big}|}{|N_i^{big}||N_{i+1}^{small}|} \qquad N_{i+1}^{small} \subseteq N_{i+1}^{big}, \text{ Eq. (23)}$$

$$\geq \frac{1}{10^{20}\ell \log k} \cdot \frac{1}{|N_{i+1}^{small}|}$$

and the claim follows.

$\square$

21

The idea behind σ is that it drops by an amount proportional to the drop in the cost $\varphi(X, C_i) - \varphi(X, C_{i+1})$. This is substantiated by the following claim.

Claim 4.9. *Assume that $10^5 k > \frac{400\varphi(X,C_i)}{|K|R_i^2}$. Then for any C_{i+1} we have*

$$\sigma_i - \overline{\sigma_{i+1}} \geq 10^{25} \ell \frac{\varphi(X, C_i) - \varphi(X, C_{i+1})}{\varphi(X, C_i)} \frac{|K|}{|N_i^{small}|}.$$

Proof. Using the assumption from the statement we get that $\sigma_i = 10^{25} \ell \log\left(\frac{400\varphi(X,C_i)}{|K|R_i^2}\right) \frac{|K|}{|N_i^{small}|}$. We have

$$
\begin{aligned}
\frac{\sigma_i - \overline{\sigma_{i+1}}}{10^{25}\ell|K|/|N_i^{small}|} &= \left(\log \frac{400\varphi(X, C_i)}{|K|R_i^2} - \log \frac{400\varphi(X, C_{i+1})}{|K|R_i^2}\right) \\
&= \log \frac{\varphi(X, C_i)}{\varphi(X, C_{i+1})} \\
&\geq \left(1 - \frac{\varphi(X, C_{i+1})}{\varphi(X, C_i)}\right) \qquad\qquad \text{\color{red}Fact 3.7} \\
&= \frac{\varphi(X, C_i) - \varphi(X, C_{i+1})}{\varphi(X, C_i)}
\end{aligned}
$$

as needed. $\qquad\square$

Note however, that $\mathrm{E}[\overline{\sigma_{i+1}} - \sigma_{i+1}]$ is not necessarily positive since $\sigma_{i+1} > \overline{\sigma_{i+1}}$ whenever $R_{i+1} \neq R_i$. In these cases, we can bound the difference $\sigma_{i+1} - \overline{\sigma_{i+1}} \leq \sigma_{i+1} = O(\ell \log k \cdot \frac{|K|}{|N_{i+1}^{small}|})$. If there is large enough drop in the size of the neighborhood, we have seen in {\color{red}Claim 4.8} that the drop in ρ can pay for the negative value of $-\sigma_{i+1}$ that we need to pay to make the left hand side of {\color{red}Eq. (19)} positive.

This is the point of the first part of the next claim. The second part argues that if we are in the case $\varphi(K, C_i) = O(\varphi(X, C_i)/k)$, we can also account for the potentially negative term $\mathrm{E}[\overline{\sigma_{i+1}} - \sigma_{i+1}]$. This time this is because in this special case, the specific value of R_i anyway does not affect the size of σ_i which is "maxed out" at value $O(\ell \log k \cdot \frac{|K|}{|N_i^{small}|})$.

Claim 4.10.   *1. Assume that $|N_{i+1}^{big}| \leq (1 - 1/(10^{20}\ell \log k)|N_i^{big}|$. Then, we have*

$$\rho_i - \rho_{i+1} + \overline{\sigma_{i+1}} - \sigma_{i+1} \geq 0.$$

2. Assume that $10^5 k \leq \frac{400\varphi(X,C_i)}{|K|R_i^2}$. Then,

$$\rho_i - \rho_{i+1} + \sigma_i - \sigma_{i+1} \geq 0.$$

Proof. First, assume that $|N_{i+1}^{big}| \leq (1 - \frac{1}{10^{20}\ell \log k})|N_i^{big}|$. In this case, we use {\color{red}Claim 4.8} to get that

$$\rho_i - \rho_{i+1} \geq 10^{30}\ell \log k \frac{|K|}{|N_{i+1}^{small}|}$$

On the other hand, we certainly have

$$\sigma_{i+1} \leq 10^{25}\ell \log(10^5 k) \cdot \frac{|K|}{|N_{i+1}^{small}|}$$

22

Hence, we have

$$\rho_i - \rho_{i+1} + \overline{\sigma_{i+1}} - \sigma_{i+1} \geq 0 \tag{24}$$

since we can assume $k \geq 2$ (for $k = 1$ there is not much to prove) and we are done.

Next, assume $10^5 k \leq \frac{400\varphi(X,C_i)}{|K|R_i^2}$. Simplifying the bound Eq. (22) from Claim 4.8 we get

$$\rho_i - \rho_{i+1} \geq 10^{50}\ell^2 \log^2 k |K| \left(\frac{1}{|N_{i+1}^{small}|} - \frac{1}{|N_i^{small}|} \right) \tag{25}$$

On the other hand, by our assumption we have $\sigma_i = 10^{25}\ell \log(10^5 k) \cdot \frac{|K|}{|N_i^{small}|}$ and it certainly has to hold that $\sigma_{i+1} \leq 10^{25}\ell \log(10^5 k) \cdot \frac{|K|}{|N_{i+1}^{small}|}$, hence we get

$$\sigma_{i+1} - \sigma_i \leq 10^{25}\ell \log(10^5 k) \cdot |K| \left(\frac{1}{|N_{i+1}^{small}|} - \frac{1}{|N_i^{small}|} \right) \tag{26}$$

Comparing Eqs. (25) and (26), we infer that $\rho_i - \rho_{i+1} + \sigma_i - \sigma_{i+1} \geq 0$, as needed. $\square$

4.4 Hard and easy cases

In this section, we formalize the "hard and easy case" from Section 2.3 and prove the necessary preparatory results for each case.

At first, we get rid of the special case when $\varphi(K, C_i) = O(\varphi(X, C_i)/k)$.

Claim 4.11. *Assume that* $10^5 k \leq \frac{400\varphi(X,C_i)}{|K|R_i^2}$. *Then Eq. (19) is satisfied.*

Proof. The condition from the statement in other words means that σ_i has the "maxed out" value of $10^{25}\ell \log(10^5 \cdot k) \frac{|K|}{|N_{i+1}^{small}|}$.

Using Claim 4.10 item (2) we infer that the left hand side of Eq. (19) can be lower bounded by

$$(\pi_i - \mathrm{E}[\pi_{i+1}]) + 0 \geq \ell/k. \tag{27}$$

On the other hand, for the right-hand side of Eq. (19) we have

$$\begin{aligned}
\frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)} &\leq \frac{\ell \cdot 2|K|R_i^2}{\varphi(X,C_i)} && \text{Claim 4.5} \\
&\leq \frac{2 \cdot 400\ell}{10^5 k} && \text{assumption}
\end{aligned} \tag{28}$$

Eqs. (27) and (28) imply that Eq. (19) holds in this case, as needed. $\square$

In the rest of the proof we only consider the case when

$$10^5 \cdot k > \frac{400\varphi(X,C_i)}{|K|R_i^2} \tag{29}$$

Consider the probability distribution over X used to sample in the current, $i + 1$th, step. That is, consider the probability space where a point $c \in X$ has probability $\varphi(c, C_i)/\varphi(X, C_i)$. Consider the random variable δ_i on this space that assigns the value $\varphi(X, C_i) - \varphi(X, C_i \cup \{c\})$ to the sampled point c. That is, δ_i is the random variable measuring the drop in the cost if we sampled just one point of X proportional to its individual cost.

We define a value ξ_i as the $1/(2\ell)$th quantile of the distribution of δ_i. Formally, ξ_i is the largest number such that

$$\mathrm{P}\left(\varphi(X, C_i) - \varphi(X, C_i \cup \{c\}) \geq \xi_i\right) \geq \frac{1}{2\ell} \tag{30}$$

Definition 4.12 (Easy and hard clusters). *We say that K is easy with respect to C_i (or in the $i + 1$th step) if and only if*

$$\xi_i < \frac{\varphi(N_i^{small}, C_i)}{1500}. \tag{31}$$

Otherwise, K is hard.

The argumentation for the easy and hard cases differs. We will next prove Claim 4.14 that we rely on in the easy case and Claim 4.15 that we rely on in the hard case.

Claim 4.13. *Any point $c \in N_i^{small}$ has the property that $\varphi(X, C_i) - \varphi(X, C_i \cup \{c\}) \geq \frac{\varphi(N_i^{small}, C_i)}{1500}$.*

Proof. Let us fix some $c \in N_i^{small}$. Consider any point $x \in N_i^{small}$. By Claim 4.5, we have $\varphi(x, C_i) \geq R_i^2/200$. On the other hand, we have

$$d(c, x) \leq d(c, \mu(K)) + d(\mu(K), x) \leq 2 \cdot R_i/100.$$

Hence, we get

$$\varphi(x, C_i) - \varphi(x, C_i \cup \{c\}) \geq \frac{R_i^2}{200} - \frac{4R_i^2}{10^4} \geq \frac{R_i^2}{500} \tag{32}$$

and summing up Eq. (32) for all $x \in N_i^{small}$, we get

$$\varphi(X, C_i) - \varphi(X, C_i \cup \{c\}) \geq \varphi(N_i^{small}, C_i) - \varphi(N_i^{small}, C_i \cup \{c\}) \tag{33}$$

$$\geq |N_i^{small}| \cdot \frac{R_i^2}{500} \qquad \text{Eq. (32)} \tag{34}$$

$$\geq \frac{\varphi(N_i^{small}, C_i)}{1500} \qquad \text{Claim 4.5 item 2} \tag{35}$$

$\square$

Claim 4.14 (Claim for the easy case). *Assume that K is easy in the $i + 1$th step. Then, for any $x \in N_i^{small}$ we have that $c_{i+1} = x$ with probability at least $\frac{\ell\varphi(x, C_i)}{2\varphi(X, C_i)}$. In particular, this implies:*

1. *$c_{i+1} \in N_i^{small}$ with probability at least $\frac{\ell\varphi(N_i^{small}, C_i)}{2\varphi(X, C_i)}$,*

2. *$c_{i+1} \in K$ with probability at least $\frac{\ell\varphi(K, C_i)}{2000\varphi(X, C_i)}$.*

Proof. Fix any $x \in N_i^{small}$. For any $1 \leq j \leq \ell$ consider the following event E_j.

Event E_j: We have $c_{i+1}^j = x$. Moreover, for every $1 \leq j' \leq \ell$ with $j' \neq j$, we have $\varphi(X, C_i) - \varphi(X, C_i \cup \{c_{j'}\}) \leq \xi_i$.

By independence of all ℓ samples of candidate centers and definition of ξ_i, we have that

$$\mathrm{P}(E_j) \geq \frac{\varphi(x, C_i)}{\varphi(X, C_i)} \cdot (1 - 1/(2\ell))^{\ell-1} \tag{36}$$

$$\geq \frac{\varphi(x, C_i)}{\varphi(X, C_i)} \cdot \left(1 - \frac{\ell-1}{2\ell}\right) \qquad \text{union bound} \tag{37}$$

$$\geq \frac{\varphi(x, C_i)}{2\varphi(X, C_i)} \tag{38}$$

Note that since K is easy, we have $\xi_i < \frac{\varphi(N_i^{small}, C_i)}{1500}$. Hence, we apply Claim 4.13 to conclude that the event E_j implies that $c_{i+1} = x$. The upper bound from K being easy also implies that all events E_j are disjoint for different j. Thus we get

$$\mathrm{P}(c_{i+1} = x) \geq \sum_{j=1}^{\ell} \mathrm{P}(E_j) \geq \frac{\ell\varphi(x, C_i)}{2\varphi(X, C_i)} \tag{39}$$

as needed.

Next, we prove the second part of the claim. The first bullet point is proven by summing up over all points $x \in N_i^{small}$:

$$\mathrm{P}(c_{i+1} \in N_i^{small}) \geq \sum_{x \in N_i^{small}} \frac{\ell\varphi(x, C_i)}{2\varphi(X, C_i)} = \frac{\ell\varphi(N_i^{small}, C_i)}{2\varphi(X, C_i)}$$

Similarly, using Claim 4.5 item 4, we conclude that

$$\mathrm{P}(c_{i+1} \in K) \geq \mathrm{P}(c_{i+1} \in K \cap N_i^{small}) \geq \sum_{x \in K \cap N_i^{small}} \frac{\ell\varphi(x, C_i)}{2\varphi(X, C_i)} \geq \frac{\ell\varphi(K, C_i)}{2000\varphi(X, C_i)}.$$

$\square$

Claim 4.15 (Claim for the hard case). *Assume K is hard and*

$$10^5 k > \frac{400\varphi(X, C_i)}{|K|R_i^2}.$$

Then,

$$\varphi(X, C_i) - \mathrm{E}_{i+1}[\varphi(X, C_{i+1})] \geq \frac{\varphi(N_i^{small}, C_i)}{3000} \geq \frac{\varphi(K, C_i)}{10^7}$$

and

$$\sigma_i - \mathrm{E}_{i+1}[\overline{\sigma_{i+1}}] \geq 10^{15} \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)}.$$

Proof. Note that by the definition of ξ_i as the $1 - 1/(2\ell)$th quantile, the probability that $\varphi(X, C_i) - \varphi(X, C_{i+1}) < \xi_i$ is at most $(1 - 1/(2\ell))^{\ell} \leq 1/3$. Hence, with probability at least $2/3$ we have $\varphi(X, C_i) - \varphi(X, C_{i+1}) \geq \xi_i$ and this implies that

$$\varphi(X, C_i) - \mathrm{E}_{i+1}[\varphi(X, C_{i+1})] \geq 2\xi_i/3.$$

25

Plugging in that K is hard (Definition 4.12), we get

$$
\begin{aligned}
\varphi(X, C_i) - \mathrm{E}_{i+1}\varphi(X, C_{i+1}) &\geq \frac{\varphi(N_i^{small}, C_i)}{3000} \\
&\geq \frac{\varphi(K, C_i)}{10^7} \qquad \text{Claim 4.5 item (4)}.
\end{aligned}
$$

Using Claim 4.9, this implies

$$
\begin{aligned}
\sigma_i - \mathrm{E}_{i+1}[\overline{\sigma_{i+1}}] &\geq 10^{25}\frac{\ell(\varphi(X, C_i) - \mathrm{E}_{i+1}[\varphi(X, C_{i+1})])}{\varphi(X, C_i)} \cdot \frac{|K|}{|N_i^{small}|} \qquad \text{Claim 4.9} \\
&\geq 10^{25}\frac{\ell\varphi(N_i^{small}, C_i)}{3000\varphi(X, C_i)} \cdot \frac{|K|}{|N_i^{small}|} \\
&\geq 10^{20}\frac{\ell\varphi(N_i^{small}, C_i)}{\varphi(X, C_i)} \cdot \frac{\varphi(K, C_i)/(2R_i^2)}{200\varphi(N_i^{small}, C_i)/R_i^2} \qquad \text{Claim 4.5} \\
&\geq 10^{15}\frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)}
\end{aligned}
$$

as needed. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $\square$

4.5 Finishing the analysis

We are now ready to do a case distinction where for each case we combine results from the previous section to verify Eq. (19).

We will first assume that

$$
|N_i^{big} \setminus N_i^{small}| \geq \frac{1}{10^{20}\ell \log k}|N_i^{big}| \tag{40}
$$

Intuitively, in this case, we are happy since there are many points in N_i^{big} that will not be present in N_{i+1}^{big}. This implies a large drop in the potential ρ via Claim 4.8 that can pay for everything.

Claim 4.16. *Assume that $10^5 k > \frac{400\varphi(X, C_i)}{|K|R_i^2}$ and $|N_i^{big} \setminus N_i^{small}| \geq \frac{1}{10^{20}\ell \log k}|N_i^{big}|$. Then Eq. (19) is satisfied.*

Proof. First, assume that K is easy. Note that if $R_i \neq R_{i+1}$ then we can use the fact that $N_{i+1}^{big} \subseteq N_i^{small}$ and our assumption to conclude that

$$
|N_{i+1}^{big}| \leq (1 - 1/(10^{20}\ell \log k))|N_i^{big}| \tag{41}
$$

Hence, we may apply the first item in Claim 4.10 and get that

$$
\rho_i - \rho_{i+1} + \sigma_i - \sigma_{i+1} \geq 0 \tag{42}
$$

When $R_i = R_{i+1}$, the same equation holds since by definition $\overline{\sigma_{i+1}} \leq \sigma_i$.

That is, the sum of all potentials always drops. We write $*$ for the event that $c_{i+1} \in K$ and compute that

$$(\pi_i - \mathrm{E}[\pi_{i+1}]) + (\rho_i - \mathrm{E}[\rho_{i+1}]) + (\sigma_i - \mathrm{E}[\sigma_{i+1}])$$
$$= \mathrm{P}(*) \left(\pi_i - \mathrm{E}[\pi_{i+1}|*]\right) + \mathrm{P}(\neg*) \left(\pi_i - \mathrm{E}[\pi_{i+1}|\neg*]\right)$$
$$+ (\rho_i - \mathrm{E}[\rho_{i+1}]) + (\sigma_i - \mathrm{E}[\sigma_{i+1}])$$
$$\geq \mathrm{P}(*) \cdot 10^{50}\ell\log k + 0 \qquad\qquad\qquad \text{Eqs. (21) and (42)}$$
$$\geq \frac{\ell\varphi(K,C_i)}{2000\varphi(X,C_i)} \cdot 10^{50}\ell\log k \qquad\qquad\qquad \text{Claim 4.14}$$
$$\geq \frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)}.$$

That is, Eq. (19) is satisfied.

Next, assume K is hard. Then, we use Claim 4.15 to get

$$\sigma_i - \mathrm{E}_{i+1}[\overline{\sigma_{i+1}}] \geq 10^{15}\frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)}.$$

Next, whenever $R_i \neq R_{i+1}$, we have necessarily $|N_{i+1}^{big}| \leq (1 - 1/(10^{20}\ell\log k))|N_i^{big}|$ by Eq. (41) and using Claim 4.10 item (2) we conclude that

$$\rho_i - \rho_{i+1} + \overline{\sigma_{i+1}} - \sigma_{i+1} \geq 0$$

If $R_i = R_{i+1}$, above equation is also clearly satisfied since in that case $\overline{\sigma_{i+1}} = \sigma_{i+1}$.

In view of the above reasoning, we get

$$(\pi_i - \mathrm{E}[\pi_{i+1}]) + (\sigma_i - \mathrm{E}[\overline{\sigma_{i+1}}]) + (\rho_i - \mathrm{E}[\rho_{i+1}]) + (\mathrm{E}[\overline{\sigma_{i+1}} - \sigma_{i+1}])$$
$$\geq 0 + 10^{15}\frac{\ell\varphi(K,C_i)}{\varphi(X,C_i)} + 0$$

and Eq. (19) is proven. $\qquad\square$

It remains to argue about the case when

$$|N_i^{big} \setminus N_i^{small}| < \frac{1}{10^{20}\ell\log k}|N_i^{big}| \tag{43}$$

In this case, we have that the two sets N_i^{big} and N_i^{small} are basically the same. This allows us to carry out the planned argument as promised in Section 2.3.

Let us define $M \subseteq N_i^{big}$ as the set of $|N_i^{big}|/(10^{20}\ell\log k)$ points of M of maximum distance to $\mu(K)$.

We observe that whenever $c_{i+1} \in N_i^{big} \setminus M$, then for each $m \in M$ we have $m \notin N_{i+1}^{big}$. This means that $c_{i+1} \in N_i^{big} \setminus M$ implies that

$$|N_{i+1}^{big}| \leq (1 - 1/(10^{20}\ell\log k)) \cdot |N_i^{big}|. \tag{44}$$

Also, since each point $x \in M$ satisfies $\varphi(x,C_i) \leq 3R_i^2$ by Claim 4.5, item (1), we infer

$$\varphi(M,C_i) \leq \frac{|N_i^{big}|}{10^{20}\ell\log k} \cdot 3R_i^2.$$

On the other hand, by Claim 4.5, item (2) and the fact that $|N_i^{small}| \geq |N_i^{big}|/2$ by Eq. (43), we have

$$\varphi(N_i^{small}, C_i) \geq |N_i^{small}|R_i^2/200 \geq |N_i^{big}|R_i^2/400.$$

Putting these two facts together, we get

$$\varphi(M, C_i) \leq \frac{3R_i^2 \cdot 400\varphi(N_i^{small}, C_i)}{10^{20}\ell \log k \cdot R_i^2} \leq \frac{\varphi(N_i^{small}, C_i)}{10^{16}\ell \log k}. \tag{45}$$

Since by Eq. (43) we have $M \supseteq N_i^{big} \setminus N_i^{small}$, we can write

$$\varphi(N_i^{big}, C_i) \leq \varphi(N_i^{small}, C_i) + \varphi(M, C_i) \leq 3\varphi(N_i^{small}, C_i)/2 \tag{46}$$

Hence, we have two results Eq. (43) and Eq. (46) that both formalize the intuition that we do not really need to distinguish between N_i^{big} and N_i^{small}. We now consider the easy and the hard case separately and finish the analysis in the following two claims.

Claim 4.17. *Assume that $10^5 k > \frac{400\varphi(X,C_i)}{|K|R_i^2}$ and $|N_i^{big} \setminus N_i^{small}| < \frac{1}{10^{20}\ell \log k}|N_i^{big}|$. Moreover, assume K is easy. Then Eq. (19) is satisfied.*

Proof. Using Claim 4.14 for the set $N_i^{big} \setminus M = N_i^{small} \setminus M$, we infer that $c_{i+1} \in N_i^{big} \setminus M$ with probability at least

$$\frac{\ell\varphi(N_i^{big} \setminus M, C_i)}{2\varphi(X, C_i)} \geq \frac{\ell\varphi(N_i^{big}, C_i)}{4\varphi(X, C_i)}$$

where we used Eq. (45).

This implies that with probability at least $\frac{\ell\varphi(N_i^{big}, C_i)}{4\varphi(X,C_i)}$ we have $\pi_{i+1} = 0$ and using Eq. (21), we get

$$\pi_i - \mathrm{E}[\pi_{i+1}] \geq \frac{\ell\varphi(N_i^{small}, C_i)}{4\varphi(X, C_i)} \cdot 10^{50}\ell \log k$$

Using Eq. (46), we infer that

$$\pi_i - \mathrm{E}[\pi_{i+1}] \geq \frac{\ell\varphi(N_i^{big}, C_i)}{10\varphi(X, C_i)} \cdot 10^{50}\ell \log k$$

Next, note that $\sigma_{i+1} \geq \sigma_i$ only in the case when $R_i \neq R_{i+1}$ and in that case we have

$$\sigma_{i+1} \leq 10^{25}\ell \log(10^5 k)\frac{|K|}{|N_{i+1}^{small}|} \leq 10^{25}\ell \log(10^5 k).$$

We have $\mathrm{P}(R_i \neq R_{i+1}) \leq \mathrm{P}(c_{i+1} \in N_i^{big}) \leq \frac{\ell\varphi(N_i^{big}, C_i)}{\varphi(X,C_i)}$. Hence,

$$\sigma_i - \mathrm{E}[\sigma_{i+1}] \geq \frac{\ell\varphi(N_i^{big}, C_i)}{\varphi(X, C_i)} \cdot (-10^{25}\ell \log(10^5 k))$$

This implies

$$
\pi_i - \mathrm{E}[\pi_{i+1}] + \rho_i - \mathrm{E}[\rho_{i+1}] + \sigma_i - \mathrm{E}[\sigma_{i+1}]
$$

$$
\geq \frac{\varphi(N_i^{big}, C_i)}{\varphi(X, C_i)} \cdot 10^{49}\ell^2 \log k + 0 - \frac{\varphi(N_i^{big}, C_i)}{\varphi(X, C_i)} \cdot 10^{25}\ell^2 \log(10^5 k)
$$

$$
\geq 10^{40} \frac{\varphi(N_i^{big}, C_i)}{\varphi(X, C_i)}
$$

$$
\geq \frac{\varphi(K, C_i)}{\varphi(X, C_i)} \hspace{5cm} \textcolor{red}{\text{Claim 4.5 item 4}}
$$

as needed.

$\square$

Claim 4.18. *Assume that* $10^5 k > \frac{400\varphi(X, C_i)}{|K|R_i^2}$ *and* $|N_i^{big} \setminus N_i^{small}| < \frac{1}{10^{20}\ell \log k}|N_i^{big}|$. *Moreover, assume K is hard. Then* Eq. (19) *is satisfied.*

Proof. We will lower bound the terms $\sigma_i - \mathrm{E}[\overline{\sigma_{i+1}}]$, $\mathrm{E}[\overline{\sigma_{i+1}} - \sigma_{i+1}]$, and $\rho_i - \mathrm{E}[\rho_{i+1}]$.

First, recall that Claim 4.15 imply

$$
\sigma_i - \mathrm{E}_{i+1}[\overline{\sigma_{i+1}}] \geq 10^{15}\ell\varphi(K, C_i)/\varphi(X, C_i). \tag{47}
$$

Next, we have that $\sigma_{i+1} > \overline{\sigma_{i+1}}$ only when $R_i \neq R_{i+1}$, which happens only when $c_{i+1} \in N_i^{big}$, otherwise we have $\sigma_{i+1} = \overline{\sigma_{i+1}}$. Also, we have $\sigma_{i+1} \leq 10^{25}\ell \log(10^5 k)\frac{|K|}{|N_{i+1}^{small}|}$, hence we get

$$
\mathrm{E}[\overline{\sigma_{i+1}} - \sigma_{i+1}] \geq \mathrm{P}(c_{i+1} \in N_i^{big}) \cdot \left(-10^{25}\ell\frac{|K|}{|N_{i+1}^{small}|} \log(10^5 k)\right). \tag{48}
$$

We rewrite the right hand side as follows. First, note that

$$
\mathrm{P}(c_{i+1} \in N_i^{big}) = \mathrm{P}(c_{i+1} \in M) + \mathrm{P}(c_{i+1} \in N_i^{big} \setminus M).
$$

We bound the first term as follows:

$$
\begin{aligned}
\mathrm{P}(c_{i+1} \in M) &\leq \mathrm{P}(\exists j : c_{i+1}^j \in M) \\
&\leq \frac{\ell\varphi(M, C_i)}{\varphi(X, C_i)} \\
&\leq \frac{\ell\varphi(N_i^{small}, C_i)}{10^{16}\ell \log k \cdot \varphi(X, C_i)} \hspace{2cm} \textcolor{red}{\text{Eq. (45)}}
\end{aligned}
$$

Thus we can continue bounding one part of the right hand side of Eq. (48) as

$$
\mathrm{P}(c_{i+1} \in M) \cdot 10^{25}\ell\frac{|K|}{|N_{i+1}^{small}|} \log(10^5 k) \tag{49}
$$

$$
\leq \frac{\varphi(N_i^{small}, C_i)}{10^{16} \log k \cdot \varphi(X, C_i)} \cdot 10^{25}\ell\frac{|K|}{|N_i^{small}|} \log(10^5 k) \tag{50}
$$

$$
\leq \frac{\varphi(N_i^{small}, C_i)}{10^{16} \log k \cdot \varphi(X, C_i)} \cdot 10^{25}\ell\frac{400\varphi(K, C_i)/R_i^2}{\varphi(N_i^{small}, C_i)/(3R_i^2)} \log(10^5 k) \quad \textcolor{red}{\text{Eq. (46) and Claim 4.5}} \tag{51}
$$

$$
\leq 10^{14}\frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} \tag{52}
$$

Putting all this together, we get

$$\mathrm{E}[\overline{\sigma_{i+1}} - \sigma_{i+1}] \geq -10^{14} \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} - \mathrm{P}(c_{i+1} \in N_i^{big} \setminus M) \cdot \left(10^{25}\ell \frac{|K|}{|N_{i+1}^{small}|} \log(10^5 k)\right). \qquad (53)$$

Finally, we bound $\rho_i - \mathrm{E}[\rho_{i+1}]$. Using Claim 4.8 and the fact that if we sample from M, we have $|N_{i+1}^{big}| \leq |N_i^{big}| - |M| \leq (1 - 1/(10^{20}\ell \log k))|N_i^{big}|$, we get

$$\rho_i - \mathrm{E}[\rho_{i+1}] \geq \mathrm{P}(c_{i+1} \in N_i^{big} \setminus M) \cdot 10^{30}\ell \log k \cdot \frac{|K|}{|N_{i+1}^{small}|} + \mathrm{P}(c_{i+1} \in M) \cdot 0 \qquad (54)$$

$$= \mathrm{P}(c_{i+1} \in N_i^{big} \setminus M) \cdot 10^{30}\ell \log k \cdot \frac{|K|}{|N_{i+1}^{small}|} \qquad (55)$$

Putting Eqs. (47), (53) and (55) together, we get

$$\pi_i - \mathrm{E}[\pi_{i+1}] + \rho_i - \mathrm{E}[\rho_{i+1}] + \sigma_i - \mathrm{E}[\overline{\sigma_{i+1}}] + \mathrm{E}[\overline{\sigma_{i+1}} - \sigma_{i+1}]$$

$$\geq 0 + \mathrm{P}(c_{i+1} \in N_i^{big} \setminus M) \cdot 10^{30}\ell \log k \cdot \frac{|K|}{|N_{i+1}^{small}|} + 10^{15}\frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)}$$

$$- 10^{14}\frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} - \mathrm{P}(c_{i+1} \in N_i^{big} \setminus M) \cdot \left(10^{25}\ell \frac{|K|}{|N_{i+1}^{small}|} \log(10^5 k)\right)$$

$$\geq \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)}$$

and Eq. (19) is proven.

$\square$

The proof of Lemma 4.7 now follows from Claims 4.11 and 4.16 to 4.18 that cover all possible cases.

5   Analysis of greedy k-means++

In this section, we prove Theorem 1.1 that we restate here for convenience. The proof relies on Lemma 2.2 proven in Section 4.

Theorem 1.1. *Greedy k-means++ (Algorithm 2) is an $O(\ell^3 \cdot \log^3 k)$-approximation algorithm, in expectation.*

We prove the theorem formally by a potential argument. We set up a potential in Definition 5.1 and track it during the algorithm. We prove in Proposition 5.2 that at the beginning the size of the potential is $O(\ell^3 \log^3 k) \cdot OPT$. At the end of the algorithm, the potential is at least as large as the cost of the final solution as proved in Proposition 5.3. Finally, in Proposition 5.7 we prove that we expect the potential only to decrease in between two steps of the algorithm. Together, these results prove Theorem 1.1.

5.1   The potential and the intuition behind it

In the rest of the section, we prove Theorem 1.1. As in the original proof of [AV07], we introduce a potential function that assigns each optimal cluster some potential.

Before we introduce it, recall Definition 4.3 where $\mathrm{HIT}_{i+1}^{j}(K)$ is defined as an indicator for whether in $i + 1$th step K is uncovered and unsolved and $c_{i+1}^{j} \in K$. We also have $\mathrm{HIT}_{i+1} = \sum_{j=1}^{\ell} \mathrm{HIT}_{i+1}^{j}$ and $\mathrm{HIT}_{\geq i+1} = \sum_{\iota=i+1}^{k} \mathrm{HIT}_{\iota}$. Also, let b_i be the number of bad steps so far where a step $i + 1$ is bad whenever c_{i+1} is a point of a cluster covered with respect to C_i. In the definition of Φ_i we condition on the randomness of the first i steps of the algorithm which makes values like b_i deterministic.

We also use the following notation: $\mathcal{K}$ is the set of all clusters of a fixed optimal solution; we have $\mathcal{K} = \mathcal{K}_i^{\mathcal{U}} \sqcup \mathcal{K}_i^{\mathcal{C}}$, i.e., we split the clusters to uncovered and covered with respect to C_i. We have $X_i^{\mathcal{U}} = \bigcup_{x \in K \in \mathcal{K}_i^{\mathcal{U}}} x$, i.e., $X_i^{\mathcal{U}}$ is the set of points in uncovered clusters, we have $X_i^{\mathcal{C}} = X \setminus X_i^{\mathcal{U}}$. Finally, we split the uncovered clusters into unsolved and solved. Formally, $\mathcal{K}_i^{\mathcal{U}} = \mathcal{K}_i^{\mathcal{U}\mathcal{U}} \sqcup \mathcal{K}_i^{\mathcal{U}\mathcal{S}}$ and $X_i^{\mathcal{U}} = X_i^{\mathcal{U}\mathcal{U}} \sqcup X_i^{\mathcal{U}\mathcal{S}}$. We do not use the notation u_i for the number of uncovered clusters as in Section 2 since this value is exactly equal to $k - i + b_i$.

We choose our potential as follows:

Definition 5.1. *Fix a step i of Algorithm 3. We define a potential Φ_i as follows.*

$$\Phi_i = \Phi_i^1 + \Phi_i^2 + \Phi_i^3 \tag{56}$$

$$= 10^{10} \ell \, (1 + H_{k-i}) \cdot \varphi(X_i^{\mathcal{C}}, C_i) \tag{57}$$

$$+ 10^{20} \ell \sum_{K \in \mathcal{K}_i^{\mathcal{U}}} (1 + E_{\geq i+1}[HIT_{\geq i+1}(K)]) \cdot (1 + H_{k-i}) \cdot \varphi^*(K) \tag{58}$$

$$+ b_i \cdot \frac{\varphi(X, C_i)}{k - i + b_i} \tag{59}$$

The intuition behind the potential is as follows. The potential function is very similar to the potential of [AV07] although our analysis is more complicated. Let us walk through the three terms $\Phi_i^1, \Phi_i^2, \Phi_i^3$ of the potential and explain the intuition behind each of them.

The first term of the potential, Φ_i^1, can be thought of as follows: every covered cluster K has potential proportional to $(1 + H_{k-i})\varphi(K, C_i)$. In the end, the cluster needs to have potential $\varphi(K, C_i)$ to pay for itself, so it already has a surplus of $H_{k-i} \cdot \varphi(K, C_i)$ of potential. This means that in the $i + 1$-th step, each covered cluster can "pay" a cost of $\varphi(K, C_i)/(k - i)$. We use this cost to pay for the fact that $i + 1$th step can be bad; formally, in that case, Φ_i^3 increases and we pay for that increase by the decrease in Φ_i^1.

The second term of the potential, Φ_i^2, has the following intuition. At the beginning, every (uncovered) cluster gets potential proportional to $E_{\geq 1}[HIT_{\geq 1}(K)] \cdot (1 + H_{k-i}) \cdot \varphi^*(K)$. In the original analysis of [AV07] it would be only $(1 + H_{k-i}) \cdot 5\varphi^*(K)$ and the aim of the potential would be that if we at some point sample from K, we use the 5 approximation result of Lemma 3.3 to argue that, in expectation, we can now change $5\varphi^*(K)$ for $\varphi(K, C_{i+1})$, which would make the potential of the newly covered cluster proportional to $(1 + H_{k-i}) \cdot \varphi(K, C_{i+1})$ which is exactly the potential that every covered cluster is supposed to have.

In our analysis, the additional term $E_{\geq 1}[HIT_{\geq 1}(K)]$ allows K to "pay" the cost $(1 + H_{k-i})\varphi(K, C_i \cup \{c_{i+1}^{j}\})$ whenever some candidate center c_{i+1}^{j} happens to be sampled from K. If $c_{i+1}^{j} = c_{i+1}$, we use the paid cost to give K enough potential as it is required being now covered. If $c_{i+1}^{j} \neq c_{i+1}$, this part of the potential that K "paid" is still subtracted from the potential of K although it remains uncovered.

One additional subtlety is that we replace H_k by H_{k-i} in the potential of every uncovered cluster. This allows us to argue that every solved uncovered cluster, i.e., every uncovered cluster with $\Phi(K, C_i) \leq 10^5 \varphi^*(K)$ also pays the cost proportional to $\Phi(K, C_i)/(k - i)$ in every round in

the same way as uncovered clusters do. We need to use this fact essentially because our random variable HIT is counting hits of a cluster only until it becomes solved. Hence, we need a small separate argument for solved clusters inside the proof.

Finally, we come to the last part of the potential, Φ_i^3. This part of the potential is paying for the fact that there were some bad steps. In [AV07], this part of the potential would be equal to $b_i \cdot \frac{\varphi(X_i^{\mathcal{U}}, C_i)}{k-i+b_i}$ and its meaning would be that it can pay for b_i "average" uncovered clusters. In the end, when $i = k$, it simply pays for all uncovered clusters. The intuition about the new problems we face here is described in Section 2.4. In summary, there is a mismatch between the optimization of $\varphi(X_{i+1}^{\mathcal{U}}, C_{i+1})$ that we wish to optimize for and $\varphi(X, C_{i+1})$ that the greedy optimizes for. While this makes the proof substantially more technical, the definition of the potential Φ_i^3 is very similar to that used by [AV07]; the only difference is that we replace the term $\varphi(X_i^{\mathcal{U}}, C_i)$ by $\varphi(X, C_i)$, essentially because the greedy rule optimizes for the latter, not the former expression.

5.2 The formal proof

In this section, we give a formal proof of Theorem 1.1. It follows from Propositions 5.2, 5.3 and 5.7.

Proposition 5.2. *We have $E[\Phi_1] = O(\ell^3 \log^3 k) \cdot OPT$.*

Proof. Let us go through the three parts of Φ_1. There was only one node sampled, hence only one covered cluster. Using Lemma 3.2, we conclude that $\mathrm{E}[\varphi(X_1^{\mathcal{C}}, C_1)] \le 2OPT$, hence $\mathrm{E}[\Phi_i^1] = O(\ell \log k) \cdot OPT$. Next, we use Lemma 2.2 to conclude that $\mathrm{E}[\mathrm{HIT}_{\ge 1}(K)] = O(\ell^2 \log^2 k)$ for every cluster K, hence $\Phi_1^2 = O(\ell^3 \log^3 k) \cdot OPT$. Finally, $b_1 = 0$ since the first center was certainly picked from an uncovered cluster, hence $\Phi_1^3 = 0$. $\qquad\square$

Proposition 5.3. *We have $\Phi_k \ge \varphi(X, C_k)$.*

Proof. For $i = k$ we have $b_i/(k - i + b_i) = 1$ and hence $\Phi_k \ge \Phi_k^3 = \varphi(X, C_k)$. $\qquad\square$

The main part of our proof of Theorem 1.1 is to show that the potential Φ_i only decreases in expectation. We prove it in Proposition 5.7 after analyzing all three parts of the potential $\Phi_i^1, \Phi_i^2, \Phi_i^3$.

Proposition 5.4. *Fix a step $i \ge 1$. We have*

$$\Delta\Phi_i^1 = \Phi_i^1 - E_{i+1}[\Phi_{i+1}^1]$$
$$\ge \frac{10^{10}\ell}{k-i} \cdot \varphi(X_i^{\mathcal{C}}, C_i)$$
$$- 10^{15}\ell \sum_{K \in \mathcal{K}_i^{\mathcal{U}\mathcal{U}}} \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K)$$
$$- 10^{15}\ell \sum_{K \in \mathcal{K}_i^{\mathcal{U}\mathcal{S}}} P(c_{i+1} \in K) \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K).$$

The intuition behind $\Delta\Phi_i^1$ is as follows. The first part is proportional to $\varphi(X^{\mathcal{C}}, C_i)/(k-i)$; this is what we are paying for the fact that the i-th step can be bad, i.e., the first term will dominate a similar, negative, term in $\Delta\Phi_i^3$. The second and the third part of the difference corresponds to the fact that some uncovered clusters can become covered and we need to ensure they have potential proportional to $(1 + H_{k-i})\varphi(K, C_{i+1})$ on them in this case. We argue differently about the solved and unsolved clusters, hence two expressions. They are accounted for by the corresponding drop in the potential $\Delta\Phi_i^2$.

Proof. We write

$$\Delta\Phi_i^1/(10^{10}\ell) = \left((1 + H_{k-i}) \cdot \varphi(X_i^{\mathcal{C}}, C_i)\right) - \mathrm{E}_{i+1}\left[(1 + H_{k-i-1}) \cdot \varphi(X_{i+1}^{\mathcal{C}}, C_{i+1})\right]$$

For every $K \in \mathcal{K}_i^{\mathcal{C}}$ we upper bound the the term $\varphi(K, C_{i+1})$ by $\varphi(K, C_i)$ in the above expression. However, notice that $X_{i+1}^{\mathcal{C}}$ potentially contains one additional newly covered cluster. We can hence write:

$$\Delta\Phi_i^1/(10^{10}\ell) \geq \frac{1}{k-i} \cdot \varphi(X_i^{\mathcal{C}}, C_i) - \sum_{K \in \mathcal{K}_i^{\mathcal{U}}} \mathrm{P}(c_{i+1} \in K) \cdot (1 + H_{k-i-1}) \cdot \mathrm{E}_{i+1}[\varphi(K, C_{i+1})|c_{i+1} \in K]$$

We split the sum on the right hand side to the summation over $K \in \mathcal{K}_i^{\mathcal{US}}$ and $K \in \mathcal{K}_i^{\mathcal{UU}}$. To bound the first part, consider any $K \in \mathcal{K}_i^{\mathcal{UU}}$ and write

$$\mathrm{P}(c_{i+1} \in K) \cdot \mathrm{E}_{i+1}[\varphi(K, C_{i+1})|c_{i+1} \in K]$$

$$= \mathrm{P}(c_{i+1} \in K) \cdot \sum_{j=1}^{\ell} \mathrm{P}(c_{i+1} = c_{i+1}^j | c_{i+1} \in K) \cdot \mathrm{E}\left[\varphi(K, C_i \cup \{c_{i+1}^j\})|c_{i+1} \in K \wedge c_{i+1} = c_{i+1}^j\right]$$

$$= \sum_{j=1}^{\ell} \mathrm{P}(c_{i+1}^j \in K \wedge c_{i+1} = c_{i+1}^j) \cdot \mathrm{E}\left[\varphi(K, C_i \cup \{c_{i+1}^j\})|c_{i+1}^j \in K \wedge c_{i+1} = c_{i+1}^j\right]$$

$$\leq \sum_{j=1}^{\ell} \mathrm{P}(c_{i+1}^j \in K \wedge c_{i+1} = c_{i+1}^j) \cdot \mathrm{E}\left[\varphi(K, C_i \cup \{c_{i+1}^j\})|c_{i+1}^j \in K \wedge c_{i+1} = c_{i+1}^j\right]$$

$$\quad + \mathrm{P}(c_{i+1}^j \in K \wedge c_{i+1} \neq c_{i+1}^j) \cdot \mathrm{E}\left[\varphi(K, C_i \cup \{c_{i+1}^j\})|c_{i+1}^j \in K \wedge c_{i+1} \neq c_{i+1}^j\right]$$

$$= \sum_{j=1}^{\ell} \mathrm{P}(c_{i+1}^j \in K) \cdot \mathrm{E}\left[\varphi(K, C_i \cup \{c_{i+1}^j\})|c_{i+1}^j \in K\right]$$

$$\leq \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} \cdot 5\varphi^*(K)$$

where we used [Lemma 3.3](#) in the last inequality.

On the other hand, for each $K \in \mathcal{K}_i^{\mathcal{US}}$ we can use the definition of solved clusters to get that

$$\mathrm{P}(c_{i+1} \in K) \cdot \mathrm{E}_{i+1}\left[\varphi(K, C_{i+1})|c_{i+1} \in K\right]$$
$$\leq \mathrm{P}(c_{i+1} \in K) \cdot \varphi(K, C_i)$$
$$\leq \mathrm{P}(c_{i+1} \in K) \cdot 10^5 \varphi^*(K).$$

$\square$

We continue with Φ_i^2.

Proposition 5.5. *Fix a step $i \geq 1$. We have*

$$\Phi_i^2 - E_{i+1}[\Phi_{i+1}^2] \geq 10^{20}\ell \sum_{K \in \mathcal{K}_i^{\mathcal{U}\mathcal{U}}} \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K)$$

$$+ 10^{20}\ell \sum_{K \in \mathcal{K}_i^{\mathcal{U}\mathcal{S}}} P(c_{i+1} \in K) \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K)$$

$$+ \frac{10^{10}\ell\varphi(X_i^{\mathcal{U}\mathcal{S}}, C_i)}{k - i}$$

The intuition behind $\Delta\Phi_i^2$ is as follows. The first part of the potential is saying that whenever a candidate center c_{i+1}^j hits an unsolved cluster K, we can pay the potential for K to become covered. The second part is saying that whenever a solved cluster K becomes covered, we can also pay the due potential; this is simple since the cost of K is already small. These two terms dominate the respective decreases in $\Delta\Phi_i^1$. Finally, each solved cluster pays a cost proportional to $\varphi^*(K)/(k-i)$ which is proportional to $\varphi(K, C_i)/(k-i)$ in every step; this is analogous to the first term of $\Delta\Phi_i^1$.

Proof. We have

$$\Delta\Phi_i^2/(10^{20}\ell) = \sum_{K \in \mathcal{K}_i^{\mathcal{U}}} (1 + E_{\geq i+1}[\mathrm{HIT}_{\geq i+1}(K)]) \cdot (1 + H_{k-i}) \cdot \varphi^*(K) \tag{60}$$

$$- E_{i+1}\left[\sum_{K \in \mathcal{K}_{i+1}^{\mathcal{U}}} (1 + E_{\geq i+2}[\mathrm{HIT}_{\geq i+2}(K)]) \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K) \right]$$

We bound this expression as follows:

$$\Delta\Phi_i^2/(10^{20}\ell) \geq \sum_{K \in \mathcal{K}_i^{\mathcal{U}\mathcal{U}}} (E_{\geq i+1}[\mathrm{HIT}_{\geq i+1}(K)] - E_{i+1}[E_{\geq i+2}[\mathrm{HIT}_{\geq i+2}(K)]]) (1 + H_{k-i-1}) \cdot \varphi^*(K)$$

$$\tag{61}$$

$$+ \sum_{K \in \mathcal{K}_i^{\mathcal{U}\mathcal{S}}} P(c_{i+1} \in K) \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K) \tag{62}$$

$$+ \frac{1}{k-i} \cdot \varphi^*(X_i^{\mathcal{U}}) \tag{63}$$

We did the following. For each cluster that is unsolved in the ith step we simply used the fact that $\mathcal{K}_{i+1}^{\mathcal{U}} \subseteq \mathcal{K}_i^{\mathcal{U}}$ and subtracted the two expressions of Eq. (60). For the solved clusters we on the other hand used that with probability $P(c_{i+1} \in K)$ we have $K \notin \mathcal{K}_{i+1}^{\mathcal{U}}$. Finally, the last term comes from the replacement of H_{k-i} in Φ_i^2 by H_{k-i-1} in Φ_{i+1}^2.

Comparing with the desired bound from the statement, we see that the second term Eq. (62) in our bound is already what it should be. For the third term Eq. (63), we first use $\varphi^*(X_i^{\mathcal{U}}) \geq \varphi^*(X_i^{\mathcal{U}\mathcal{S}})$ and then, by definition of solved clusters, $\varphi^*(X_i^{\mathcal{U}\mathcal{S}}) \geq \frac{1}{10^5}\varphi(X_i^{\mathcal{U}\mathcal{S}}, C_i)$.

It remains to deal with the first term Eq. (61). Consider any cluster $K \in \mathcal{K}_i^{\mathcal{U}}$ that is also not solved. Then we have $E_{i+1}[\mathrm{HIT}_{i+1}^j(K)] = \frac{\varphi(K, C_i)}{\varphi(X, C_i)}$ for any $1 \leq j \leq \ell$ and we can hence compute

34

that

$$
\begin{aligned}
&\mathrm{E}_{\geq i+1}[\mathrm{HIT}_{\geq i+1}(K)] - \mathrm{E}_{i+1}\left[\mathrm{E}_{\geq i+2}[\mathrm{HIT}_{\geq i+2}(K)]\right] \\
&= \mathrm{E}_{\geq i+1}[\mathrm{HIT}_{\geq i+1}(K)] - \mathrm{HIT}_{\geq i+2}(K)]] \\
&= \mathrm{E}_{\geq i+1}[\sum_{j=1}^{\ell} \mathrm{HIT}_{i+1}^{j}] \\
&= \frac{\ell \varphi(K, C_i)}{\varphi(X, C_i)}.
\end{aligned}
$$

which concludes the proof. $\qquad\square$

We finish with the third part of the potential, Φ_i^3.

Proposition 5.6. *Fix a step $i \geq 1$. We have*

$$
\Phi_i^3 - \mathrm{E}_{i+1}[\Phi_{i+1}^3] \geq -\frac{2\ell\varphi(X_i^{\mathcal{C}} \cup X_i^{\mathcal{US}}, C_i)}{k-i} \tag{64}
$$

$$
-5 \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi(K, C_i)}{\varphi(X, C_i)} \varphi^*(K) \tag{65}
$$

The intuition behind $\Delta\Phi_i^3$ is as follows. In the original k-means++ analysis, we would here want to prove that $\Delta\Phi_i^3 \geq -\varphi(X_i^{\mathcal{C}}, C_i)/(k-i)$ where the right-hand side corresponds to the probability of a bad step multiplied by the cost of average uncovered cluster. In our setting, we first lose an additional ℓ-factor since the probability of having a bad step is ℓ times larger. We also lose a few more error terms as discussed in Section 5.1, the important part is that they can be paid for by $\Delta\Phi_i^1 + \Delta\Phi_i^2$.

Proof. We will bound $\mathrm{E}_{i+1}[\Phi_{i+1}^3] - \Phi_i^3$ instead of $\Phi_i^3 - \mathrm{E}_{i+1}[\Phi_{i+1}^3]$ to make the relevant terms positive.

At first we note that we surely know that

$$
\Phi_{i+1}^3 \leq (b_i+1)\frac{\varphi(X, C_i)}{k-(i+1)+(b_i+1)} = (b_i+1)\frac{\varphi(X, C_i)}{k-i+b_i} \tag{66}
$$

This follows by bounding $\varphi(X, C_{i+1}) \leq \varphi(X, C_i)$ and noting that the value of Φ_{i+1} is larger whenever $b_{i+1} = b_i + 1$ as opposed to $b_{i+1} = b_i$. To see it formally, we note that for any $b_i > 0$ and any $k-(i+1) \geq 0$ the inequality $\frac{b_i}{k-(i+1)+b_i} \leq \frac{b_i+1}{k-(i+1)+(b_i+1)}$ is equivalent to $\frac{k-(i+1)}{k-(i+1)+b_i} \geq \frac{k-(i+1)}{k-(i+1)+(b_i+1)}$.

We start by analysing the special case when $\varphi(X_i^{\mathcal{U}}, C_i) \leq \varphi(X_i^{\mathcal{C}}, C_i)$. In this case, we simply use this assumption and Eq. (66) to bound

$$
\mathrm{E}_{i+1}[\Phi_{i+1}^3] - \Phi_i^3 \tag{67}
$$

$$
\leq (b_i+1)\frac{\varphi(X, C_i)}{k-i+b_i} - b_i\frac{\varphi(X, C_i)}{k-i+b_i} \tag{68}
$$

$$
= \frac{\varphi(X, C_i)}{k-i+b_i} \leq 2\frac{\varphi(X_i^{\mathcal{C}}, C_i)}{k-i+b_i} \leq 2\frac{\varphi(X_i^{\mathcal{C}}, C_i)}{k-i} \tag{69}
$$

and we are done as this term is dominated by the right hand side of Eq. (64).

Next, we assume

$$
\varphi(X_i^{\mathcal{U}}, C_i) \geq \varphi(X_i^{\mathcal{C}}, C_i) \tag{70}
$$

35

We start by writing

$$\mathrm{E}_{i+1}[\Phi_{i+1}^3] - \Phi_i^3 \tag{71}$$

$$= \mathrm{E}_{i+1}\left[b_{i+1}\frac{\varphi(X, C_{i+1})}{k - (i+1) + b_{i+1}}\right] - b_i \cdot \frac{\varphi(X, C_i)}{k - i + b_i} \tag{72}$$

$$\leq \mathrm{P}(c_{i+1} \in X_i^{\mathcal{C}} \cup X_i^{\mathcal{US}})\left((b_i + 1)\frac{\varphi(X, C_i)}{k - i + b_i} - b_i\frac{\varphi(X, C_i)}{k - i + b_i}\right) \tag{73}$$

$$+ \mathrm{P}(c_{i+1} \in X_i^{\mathcal{UU}}) \cdot \left(b_i\frac{\mathrm{E}_{i+1}[\varphi(X, C_{i+1})|c_{i+1} \in X_i^{\mathcal{UU}}]}{k - (i+1) + b_i} - b_i\frac{\varphi(X, C_i)}{k - i + b_i}\right) \tag{74}$$

That is, we distinguish two cases based on where the center c_{i+1} is picked from. In the first case we pessimistically bound Φ_{i+1}^3 using Eq. (66), while in the second case we use the fact that $c_{i+1} \in X_i^{\mathcal{UU}}$ implies that $b_{i+1} = b_i$.

To bound the first term, i.e. Eq. (73), we first use that

$$\mathrm{P}(c_{i+1} \in X_i^{\mathcal{C}} \cup X_i^{\mathcal{US}}) \leq \mathrm{P}(\exists j : c_{i+1}^j \in X_i^{\mathcal{C}} \cup X_i^{\mathcal{US}}) \leq \frac{\ell\varphi(X_i^{\mathcal{C}} \cup X_i^{\mathcal{US}}, C_i)}{\varphi(X, C_i)}.$$

We also have

$$(b_i + 1)\frac{\varphi(X, C_i)}{k - i + b_i} - b_i\frac{\varphi(X, C_i)}{k - i + b_i} \leq \frac{\varphi(X, C_i)}{k - i}$$

Hence, the value of Eq. (73) is at most

$$\frac{\ell\varphi(X_i^{\mathcal{C}} \cup X_i^{\mathcal{US}}, C_i)}{\varphi(X, C_i)} \cdot \frac{\varphi(X, C_i)}{k - i} \leq \frac{\ell\varphi(X_i^{\mathcal{C}} \cup X_i^{\mathcal{US}}, C_i)}{k - i}$$

Hence, the first term, corresponding to the case when the step is bad, is conveniently dominated by Eq. (64).

In the rest of the proof, we analyze the second term Eq. (74) that corresponds to the drift of the size of the average uncovered cluster.

We start by proving that

$$\mathrm{E}_{i+1}[\varphi(X, C_{i+1})|c_{i+1} \in X_i^{\mathcal{UU}}] \leq \mathrm{E}_{i+1}[\varphi(X, C_i \cup \{c_{i+1}^1\})|c_{i+1}^1 \in X_i^{\mathcal{UU}}] \tag{75}$$

That is, we claim that if we reveal that the center c_{i+1} taken by the greedy rule is from $X_i^{\mathcal{UU}}$, we know that the expected new cost $\varphi(X, C_{i+1})$ is smaller than if we simply sampled some candidate center c_{i+1}^j and revealed it is sampled from $X_i^{\mathcal{UU}}$. Eq. (75) then allows us to analyze further only the expression on its right-hand side that does not rely anymore on the greedy rule.

Eq. (75) follows from the fact our rule is greedy; to formally verify it holds, let us first write

$$\mathrm{E}_{i+1}[\varphi(X, C_{i+1})|c_{i+1} \in X_i^{\mathcal{UU}}] = \sum_{I \in \mathcal{I}} \mathrm{P}(I|c_{i+1} \in X_i^{\mathcal{UU}}) \cdot \mathrm{E}_{i+1}[\varphi(X, C_{i+1})|I] \tag{76}$$

where $\mathcal{I}$ is the set of all of at most $2^\ell \cdot \ell$ possible following revelations: For each $1 \leq j \leq \ell$, we reveal whether $c_{i+1}^j \in X_i^{\mathcal{UU}}$, and we also reveal for which index j_0 we have $c_{i+1} = c_{i+1}^{j_0}$. Notice that on the right hand side of Eq. (76) we used $\mathrm{E}_{i+1}[\varphi(X, C_{i+1})|I] = \mathrm{E}_{i+1}[\varphi(X, C_{i+1})|I \wedge c_{i+1} \in X_i^{\mathcal{UU}}]$ since $c_{i+1} \in X_i^{\mathcal{UU}}$ is always either implied by I or it is incompatible with it and then $\mathrm{P}(I|c_{i+1} \in X_i^{\mathcal{UU}}) = 0$.

Fixing any revelation I with $c_{i+1} = c_{i+1}^{j_0}$, we observe that

$$\mathrm{E}_{i+1}[\varphi(X, C_{i+1})|I] \leq \mathrm{E}_{i+1}[\varphi(X, C_i \cup \{c_{i+1}^1\})|c_{i+1}^1 \in X_i^{\mathcal{UU}}] \tag{77}$$

To see this, first rewrite the equation equivalently as

$$\mathrm{E}_{i+1}[\varphi(X, C_i \cup \{c_{i+1}^{j_0}\})|I] \le \mathrm{E}_{i+1}\left[\varphi(X, C_i \cup \{c_{i+1}^{j_0}\})|c_{i+1}^{j_0} \in X_i^{\mathcal{UU}}\right]. \tag{78}$$

Observe that the information I can be viewed as describing distributions from which all candidate centers c_{i+1}^j are sampled from, independently, together with the information that after candidate centers were sampled, it happened that $\varphi(X, C_i \cup \{c_{i+1}^{j_0}\}) \le \varphi(X, C_i \cup \{c_{i+1}^{j}\})$ for any $j \ne j_0$. This means that the correctness of Eq. (78) follows from Lemma 3.9. Plugging Eq. (77) to Eq. (76) proves Eq. (75).

We now continue with analysing the term $\mathrm{E}_{i+1}[\varphi(X, C_i \cup \{c_{i+1}^1\})|c_{i+1}^1 \in X_i^{\mathcal{UU}}]$ from Eq. (75) even further. We write:

$$\mathrm{E}_{i+1}[\varphi(X, C_i \cup \{c_{i+1}^1\})|c_{i+1}^1 \in X_i^{\mathcal{UU}}] \tag{79}$$

$$\le \varphi(X, C_i) - \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi(K, C_i)}{\varphi(X_i^{\mathcal{UU}}, C_i)} \cdot \left(\varphi(K, C_i) - \mathrm{E}_{i+1}\left[\varphi\left(K, C_i \cup \{c_{i+1}^1\}|c_{i+1}^1 \in K\right)\right]\right) \tag{80}$$

$$\le \varphi(X, C_i) - \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi(K, C_i)}{\varphi(X_i^{\mathcal{UU}}, C_i)} \cdot \left(\varphi(K, C_i) - 5\varphi^*(K)\right) \qquad \text{Lemma 3.3} \tag{81}$$

$$= \varphi(X, C_i) - \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi^2(K, C_i)}{\varphi(X_i^{\mathcal{UU}}, C_i)} + \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi(K, C_i)}{\varphi(X_i^{\mathcal{UU}}, C_i)} \cdot 5\varphi^*(K) \tag{82}$$

$$\le \varphi(X, C_i) - \frac{\varphi(X_i^{\mathcal{UU}})}{|\mathcal{K}_i^{\mathcal{UU}}|} + \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi(K, C_i)}{\varphi(X_i^{\mathcal{UU}}, C_i)} \cdot 5\varphi^*(K) \tag{83}$$

where the last bound follows from the Cauchy-Schwartz inequality (or AK inequality) $\sum_{i=1}^n x_i^2 \ge \left(\sum_{i=1}^n x_i\right)^2 / n$.

It is time to reap the fruits of our work. We plug in the bounds from Eqs. (75) and (83) to the term Eq. (74) and bound $\mathrm{P}(c_{i+1} \in X_i^{\mathcal{UU}}) \le 1$ there to conclude that

$$\text{Eq. (74)} \le \left(b_i \frac{\mathrm{E}_{i+1}[\varphi(X, C_{i+1})|c_{i+1} \in X_i^{\mathcal{UU}}]}{k - (i+1) + b_i} - b_i \frac{\varphi(X, C_i)}{k - i + b_i}\right) \tag{84}$$

$$\le b_i \cdot \left(\frac{\varphi(X, C_i) - \frac{\varphi(X_i^{\mathcal{UU}})}{|\mathcal{K}_i^{\mathcal{UU}}|} + \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi(K, C_i)}{\varphi(X_i^{\mathcal{UU}}, C_i)} \cdot 5\varphi^*(K)}{k - (i+1) + b_i} - \frac{\varphi(X, C_i)}{k - i + b_i}\right) \tag{85}$$

This can be further simplified to

$$\text{Eq. (74)} \le b_i \cdot \left(\frac{\varphi(X, C_i) - \frac{\varphi(X_i^{\mathcal{UU}})}{|\mathcal{K}_i^{\mathcal{UU}}|}}{k - (i+1) + b_i} - \frac{\varphi(X, C_i)}{k - i + b_i}\right) + \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi(K, C_i)}{\varphi(X_i^{\mathcal{UU}}, C_i)} \cdot 5\varphi^*(K) \tag{86}$$

Note that the last term of the right-hand side is already equal to Eq. (65) so to finish we need to

37

analyze the first term of the right-hand side. We do it as follows.

$$b_i \cdot \left(\frac{\varphi(X, C_i) - \frac{\varphi(X_i^{\mathcal{UU}})}{|\mathcal{K}_i^{\mathcal{UU}}|}}{k - (i+1) + b_i} - \frac{\varphi(X, C_i)}{k - i + b_i} \right) \tag{87}$$

$$\leq b_i \cdot \left(\frac{\varphi(X, C_i) - \frac{\varphi(X_i^{\mathcal{UU}})}{k - i + b_i}}{k - (i+1) + b_i} - \frac{\varphi(X, C_i)}{k - i + b_i} \right) \qquad |\mathcal{K}_i^{\mathcal{UU}}| \leq |\mathcal{K}_i^{\mathcal{U}}| \tag{88}$$

$$= b_i \cdot \left(\frac{\varphi(X_i^{\mathcal{UU}}, C_i) - \frac{\varphi(X_i^{\mathcal{UU}})}{k - i + b_i}}{k - (i+1) + b_i} - \frac{\varphi(X_i^{\mathcal{UU}}, C_i)}{k - i + b_i} \right) + b_i \varphi(X_i^{\mathcal{US}} \cup X_i^{\mathcal{C}}) \left(\frac{1}{k - (i+1) + b_i} - \frac{1}{k - i + b_i} \right) \tag{89}$$

$$= 0 + \frac{b_i \varphi(X_i^{\mathcal{US}} \cup X_i^{\mathcal{C}})}{(k - i + b_i)(k - i + b_i - 1)} \tag{90}$$

$$\leq \frac{\varphi(X_i^{\mathcal{US}} \cup X_i^{\mathcal{C}})}{k - i} \qquad i \leq k - 1 \tag{91}$$

Plugging back to Eq. (86) and all the way back to Eq. (71) finishes the proof.

$\square$

Proposition 5.7. *Fix a step $i > 1$. We have*

$$\Phi_i \geq E_{i+1}[\Phi_{i+1}].$$

Proof. Putting all bounds of Propositions 5.4 to 5.6 together, we get

$$\Phi_i - E_{i+1}[\Phi_{i+1}] = \Delta\Phi_i^1 + \Delta\Phi_i^2 + \Delta\Phi_i^3 \tag{92}$$

$$\geq \frac{10^{10}\ell}{k - i} \cdot \varphi(X_i^{\mathcal{C}}, C_i) \tag{93}$$

$$- 10^{15}\ell \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K) \tag{94}$$

$$- 10^{15}\ell \sum_{K \in \mathcal{K}_i^{\mathcal{US}}} P(c_{i+1} \in K) \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K) \tag{95}$$

$$+ 10^{20}\ell \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\ell\varphi(K, C_i)}{\varphi(X, C_i)} \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K) \tag{96}$$

$$+ 10^{20}\ell \sum_{K \in \mathcal{K}_i^{\mathcal{US}}} P(c_{i+1} \in K) \cdot (1 + H_{k-i-1}) \cdot \varphi^*(K) \tag{97}$$

$$+ \frac{10^{10}\ell\varphi(X_i^{\mathcal{US}}, C_i)}{k - i} \tag{98}$$

$$- \frac{2\ell\varphi(X_i^{\mathcal{C}} \cup X_i^{\mathcal{US}}, C_i)}{k - i} \tag{99}$$

$$- 5 \sum_{K \in \mathcal{K}_i^{\mathcal{UU}}} \frac{\varphi(K, C_i)}{\varphi(X, C_i)} \varphi^*(K) \tag{100}$$

$$\geq 0 \tag{101}$$

$\square$

38

6 A hard instance for greedy k-means++

In this section we provide a construction of a (weighted) point set where Algorithm 2 returns a solution with approximation $\Omega(\frac{\ell^3 \log^3 k}{\log^2(\ell \log k)})$ with constant probability. Formally, we prove Theorem 1.2 that we restate here for convenience.

Theorem 1.2. *For every k and $\ell \leq k^{0.1}$, there exists a point set $X \subseteq \mathbb{R}^d$ for some $d \in \mathbb{N}$ where Algorithm 2 outputs $\Omega(\ell^3 \log^3 k / \log^2(\ell \log k))$ approximate solution with constant probability.*

Recall that we already gave an informal description in Section 2.5. We first describe the point set in Section 6.1. We then give the formal analysis of greedy k-means++ on the point set in Section 6.2.

6.1 The point set

We start by describing the weighted point set X. In fact, we define the full input instance $(X, C_0, \widetilde{k})$ where C_0 is the starting set of centers (see Lemma 3.4).

We set

$$t = \frac{\ell \log k}{1000 \log(\ell \log k)}.$$

In the follow-up discussion, we always assume that k is large enough and the expressions like the one above that defines t are integers. This is for the purpose of readability; it is simple to make the proof work by adding $\lfloor \rfloor$ to all definitions that require integer values.

Recall that in the statement of Theorem 1.2 we assume that $\ell < k^{0.1}$; we did not try to optimize this bound but note that there has to be some since for $\ell \gg k$ we have $\ell^2 \log^2 k \gg \ell \cdot k$ where the right-hand side is the trivial bound on the number of hits. Note that the lower bound $\Omega(\ell \log k)$ of [BERS20] holds also for large ℓ, hence for $\ell > k^{0.1}$ the greedy k-means++ algorithm already necessarily has bad, polynomial, approximation guarantee.

We will now describe the point set X, the weights of the points, and the distances between some pairs of points. Then, we discuss how exactly we embed the points in the Euclidean space. We next list points of X (the picture to have in mind is Fig. 3).

1. There is a point b for which we have $w(b) = \frac{1}{t}$.

2. A point c is at distance 1 from b. We have $w(c) = 1/10$.

3. We have a set of points $N = \{n_1, n_2, \ldots, n_{t+1}\}$ and $M = \{m_1, \ldots, m_t\}$ defined as follows. We have $d(b, n_i) = k^i$ and $w(n_i) = w(m_i) = \frac{1000}{t}$. Each m_i lies at distance $10tk^i$ from n_i. We put the point n_{t+1} to C_0, that is, we assume that point is already sampled at the beginning.

4. We have $A = \{a_1, \ldots, a_{k^{1.2}}\}$ at distance k from c. The weight of each a_i is $\frac{\ell \log k}{k^2}$ so their total weight is $\frac{\ell \log k}{k^{0.8}}$.

5. We have $E = \{e_0, \bigcup_{e \in E_1} e, \ldots, \bigcup_{e \in E_t} e\}$ where $E_i = \{e_{i,1}, \ldots, e_{i,\sqrt{k}}\}$. Each point in $E \setminus \{e_0\}$ has distance 1 from a point e_0 which is at distance k^{2t} from b. We include e_0 to C_0. Since we also included $n_{t+1} \in C_0$ and we chose $d(b, e_0)$ large enough, it will never happen that a closest center to a point in E is in $X \setminus E$ or vice versa. Each $e_{i,j} \in E_i$ has the same weight w_i. This parameter needs to be set up quite precisely depending on the rest of the instance so we define it only later.

The number of points in this point set X is equal to $|X| = 1 + 1 + (t+1) + t + k^{1.2} + (1 + t \cdot \sqrt{k}) = O(k^{1.2})$. Two points of X, n_{t+1} and d, are already in C_0. We choose the number $\widetilde{k} = |X| - |C_0| - 1$. We will work with the input instance $(X, C_0, \widetilde{k})$. That is, in this instance, the optimal solution (Lemma 3.5) as well as the solution of greedy k-means++ selects as centers all points of X, except for one.

Arrangement: We next specify fully how to embed the point set X to the Euclidean space. So far, we only specified distances of pairs $(n_i, m_i), (b, n_i), (b, c), (c, a_i), (e_0, e_{i,j}), (e_0, b)$ for all i, j. These distances define a tree metric that we will simulate. Unfortunately, we cannot simulate exactly this tree metric in a Euclidean space, but we can come sufficiently close to it, using Fact 3.6.

We now describe the configuration. In view of Fact 3.6, vectors $(b, c), (b, n_1), \ldots, (b, n_{t+1})$ are chosen as vertices of a $(t+1)$-dimensional simplex. Each m_i lies on the ray (b, n_i).

To give an example how this embedding simulates the idealized tree metric up to $1/t$ loss, let us verify that $d(n_{i'}, n_i) = d(n_i, b) + \Theta(d(b, n_{i'})/t)$:

$$d(n_{i'}, n_i)^2 = d(n_{i'}, b)^2 + d(b, n_i)^2 + 2d(n_{i'}, b)d(n_i, b)/(t+1) = k^{2i'} + k^{2i} + \frac{2}{t+1}k^{i+i'} \quad (102)$$

where we used the cosine law and Fact 3.6. That is, we have

$$d(n_{i'}, n_i) \geq \sqrt{k^{2i} + 2k^{i+i'}/(t+1)} = k^i\sqrt{1 + 2k^{i'-i}/(t+1)} \geq k^i(1 + k^{i'-i}/(2t)) = k^i + k^{i'}/(2t) \quad (103)$$

Similarly, we can get the following bounds

$$d(m_{i'}, n_i) \geq k^i + k^{i'}/(2t) \quad (104)$$

$$d(c, n_i) \geq k^i + 1/(2t) \quad (105)$$

$$d(c, m_i) \geq (10t+1)k^i + 1/(2t) \quad (106)$$

Next, we specify the directions of the points A. Each a_i goes partly in the direction of the ray b, c and partly in a direction orthogonal to everything else. Namely, the ray (c, a_i) has direction $\frac{1}{2}(b,c) + \frac{1}{2}\nu_i$ where ν_i is orthogonal to the span of $X \setminus \{a_i\}$. Hence, the projection of a_i to (b, c) is at distance $k/\sqrt{2}$ from c. Also, for any $i < j$ we can project to the plane defined by ν_i, ν_j to see that

$$d(a_i, a_j) = k \quad (107)$$

Finally, using cosine law, we get

$$d(b, a_i)^2 = d(c, b)^2 + d(c, a_i)^2 - d(b, c)d(c, a_i)\cos 135° \quad (108)$$
$$= 1 + d(c, a_i)^2 + d(c, a_i)/\sqrt{2} \quad (109)$$
$$\geq d(c, a_i)^2 + d(c, a_i)/2 \quad (110)$$
$$= k^2 + k/2 \quad (111)$$

Finally, each vector $(e_0, e_{i,j})$ is orthogonal to the span of $X \setminus \{e_{i,j}\}$. In particular, we have

$$d(e_{i,j}, e_{i',j'}) \geq d(e_0, e_{i,j}) \quad (112)$$

for every i, j, i', j'.

Precise definition of w_i: It remains to define w_i. We define $S_i = \{b, c, a_j, n_{\leq i}, m_{\leq i}\}$. Then, we define

$$\Delta(b) = \varphi(S_i, \{n_{i+1} \cup b\}) - \varphi(S_i, \{n_{i+1}\}) \tag{113}$$

and

$$\Delta(c) = \varphi(S_i, \{n_{i+1} \cup c\}) - \varphi(S_i, \{n_{i+1}\}) \tag{114}$$

We define

$$w_i = \frac{\Delta(b) + \Delta(c)}{2}. \tag{115}$$

That is, w_i is set up so that, under some assumptions about what centers are already taken (e.g. n_{i+1} is but points of S_i are not), the drop resulted by taking $e_{i,j}$ as a center is smaller than the drop when we take b but bigger than the drop when we take c (we are yet to prove that $\Delta(b) > \Delta(c)$). Note that w_i satisfies

$$k^{2i+2} \leq w_i \leq 3000 k^{2i+2} \tag{116}$$

The lower bound follows from $\varphi(c, \{n_{i+1}\}) \geq k^{2i+2}$ using Eq. (105). The upper bound follows from the fact that $w(S_i)$ is dominated by the cost of N_i and M_i and for any $x \in S_i$ we have $d(n_{i+1}, x) < 1.1 k^{i+1}$ which follows from looking at Fig. 3.

This concludes the description of the point set X.

Remark 6.1. *Although our point set is weighted, we can make it unweighted by scaling all the weights up by a sufficiently large number and rounding them to the nearest integer.*

In fact, we believe, but do not prove, that all weights and positions of points in X from Theorem 1.2 can be made integers of order $k^{O(\log k)}$. We do not attempt a formal proof since that would require tedious arguments about rounding errors.

We note that in view of the $O(\ell^{O(1)} \log \frac{OPT(1)}{OPT(k)})$ upper bound sketched in Appendix C, the size of point weights and positions cannot be both improved to $k^{O(1)}$. Namely, for constant ℓ, any instance where Algorithm 2 is $\Omega(\log^3 k / \operatorname{poly} \log \log k)$ approximate needs to satisfy

$$\log \frac{OPT(1)}{OPT(k)} \geq \log^2 k / \operatorname{poly} \log \log k.$$

Hence, whenever $OPT(k)$ is a positive integer, we get that necessarily

$$OPT(1) = \varphi(X, \mu(X)) \geq k^{\log k / \operatorname{poly} \log \log k}.$$

6.2 Analysis of greedy k-means++ on the hard point set

In this subsection, we give the formal proof of Theorem 1.2.

First epoch:

We define the first epoch formally as the first $\widetilde{k} - k^{1.2}$ steps of Algorithm 2 (cf. Section 2.5 for the intuition behind the first epoch). This means our aim is to prove the following claim.

Claim 6.2. *After running Algorithm 2 on the instance $(X, \widetilde{k}, C_0)$ for $\widetilde{k} - k^{1.2}$ steps, with positive probability we have*

$$C_{\widetilde{k} - k^{1.2}} = X \setminus (A \cup \{c\})$$

41

We split the epoch into t phases that we, for notational reasons, index in a decreasing order as $i = t, t-1, \ldots, 1$. Our main task is to prove that in each i-th phase the point b is selected as a center with probability $\Omega(1/t)$. The i-th phase is formally defined as follows. With the exception of the very first phase, it starts when the last point of E_{i+1} is taken as a center. Alternatively, we say that a phase finishes whenever b is taken.

As a first claim, we prove that whenever b is selected as a center, with constant probability, we finish the first epoch as intended.

Claim 6.3. *Assume that in some step ι_0 during the first phase we have $b \in C_{\iota_0}$ and $(\{c\} \cup A) \cap C_{\iota_0} = \emptyset$. Then, with probability at least $1/2$, the first phase finishes with*

$$C_{\widetilde{k} - k^{1.2}} = X \setminus (A \cup \{c\}).$$

Proof. First, we upper bound the total cost of $\{c\} \cup A$ in every step $\iota \geq \iota_0$ assuming $(\{c\} \cup A) \cap C_\iota = \emptyset$. We have $\varphi(c, b) = 1 \cdot 1^2 = 1$ and $\varphi(A, b) = k^{1.2} \cdot \frac{\ell \log k}{k^2} \cdot k^2 = \ell k^{1.2} \log k$. That is, $\varphi(\{c\} \cup A, C_\iota) \leq \ell k^{1.3}$.

On the other hand, we claim that any point $x \in N \cup M \cup E$ has always cost $\varphi(x, C_\iota) \geq k^2/t$, unless $x \in C_\iota$. For the points of $N \cup M$, this is because their weight is $1000/t$ and the distance to the closest other point of X is always at least k. For the points of E, this is because the distance to the closest already taken point is always 1 (this is the point $e_0 \in C_0 \subseteq C_\iota$) and the smallest weight of a point in E is at least k^4 by Eq. (116).

In view of the above computations, we have that the probability we sample a candidate center from $\{c\} \cup A$ in step ι is at most

$$\ell \cdot \frac{\ell k^{1.3}}{(\widetilde{k} - k^{1.2} - \iota) \cdot k^2/t} \leq \frac{1}{(\widetilde{k} - k^{1.2} - \iota) \cdot k^{0.4}}.$$

Union bounding over all $\widetilde{k} < k^2$ step leads to a harmonic series summing up to $O(\log k)/k^{0.4} < 1/2$, as needed. $\qquad \square$

Next, let us analyze one phase of the first epoch. Recall that the ith phase starts after step ι for which $E_{i+1} \subseteq C_\iota$ and finishes when $E_i \subseteq C_\iota$ or $b \in C_\iota$. In the next claim, we compare the cost drops of various points with the "baseline cost drop" of taking a point in E_i.

Claim 6.4. *Assume that $N_{\geq i+1} \cup M_{\geq i+2} \cup E_{\geq i+1} \subseteq C_\iota$ while $(N_{\leq i} \cup M_{\leq i} \cup E_{\leq i} \cup A \cup \{b, c\}) \cap C_\iota = \emptyset$. Let $\Delta(x) = \varphi(X, C_\iota) - \varphi(X, C_\iota \cup \{x\})$ be a cost drop of a point $x \in X$. Then, we have for any $i' < i$ and any j we have*

$$\Delta(n_{i'}), \Delta(m_{i'}), \Delta(c) < \Delta(e_{i,j}) < \Delta(b) < \Delta(n_i), \Delta(m_{i+1})$$

where $e_{i,j}$ is arbitrary point not in C_ι.

Proof. First, we prove that $\Delta(b) > \Delta(c)$. For $x \in \{c\} \cup A$ we have $\varphi(x, c) \leq \varphi(x, b)$, whereas for any $x \in \{b\} \cup N_{\leq i} \cup M_{\leq i+1}$ we have $\varphi(x, b) \leq \varphi(x, c)$.

So, we first upper bound the drop difference for $\{c\} \cup A$:

$$\varphi(c \cup A, b) - \varphi(c \cup A, c) = 1 + \varphi(A, b) - \varphi(A, c)$$
$$= 1 + k^{1.2} \cdot \frac{\log k}{k^2} \cdot (1 + d(c, a_1)/\sqrt{2})$$
$$= O(k^{0.2} \log k)$$

where we used Eq. (109) and $d(c, a_1) = k$.

42

On the other hand, consider just the point n_1. Using the cosine law, we have

$$\varphi(n_1, b) - \varphi(n_1, c) \geq 2d(n_1, b)d(b, c)/t \geq k/t$$

That is, the difference in the cost drop at point n_1 dominates all other points where c has a larger cost drop than b. This means that $\Delta(b) > \Delta(c)$ and by definition of w_i, we already get for any j that

$$\Delta(c) < \Delta(e_{i,j}) < \Delta(b).$$

Next, consider the point n_i. We will prove that $\Delta(n_i) > \Delta(b)$. The intuitive reason for this is that m_i is sufficiently far away to make the drop difference between n_i and b there dominate the other terms. Concretely, we have

$$\varphi(m_i, b) - \varphi(m_i, n_i) = 1000/t \cdot \left(((10t+1)k^i)^2 - (10tk^i)^2\right) \geq 1000/t \cdot 20tk^{2i} \geq 20000k^{2i} \quad (117)$$

On the other hand, consider the set $T = \{b, c\} \cup A \cup N_{<i} \cup M_{<i}$ of points x for which $\varphi(x, n_i) \geq \varphi(x, b)$ (note that $\varphi(m_{i+1}, C_\iota \cup \{b\}) = \varphi(m_{i+1}, C_\iota \cup \{c\}) = \varphi(m_{i+1}, n_{i+1})$, that is, m_{i+1} is not enjoying any cost drop). Using the fact that the largest distance $d(x, n_i)$ for $x \in T$ is $10tk^{i-1}$ for $x = m_{i-1}$, and the fact that $w(T) \leq 3000$, we get

$$\varphi(T, n_i) - \varphi(T, b) \leq \varphi(T, n_i) \leq w(T) \cdot (10tk^{i-1} + k^i)^2 \leq 3000 \cdot (2k^i)^2 = 12000k^{2i} \quad (118)$$

Comparing with Eq. (117), we conclude that $\Delta(n_i) > \Delta(b)$ as needed.

Next, consider the point m_{i+1}. We have $\Delta(m_{i+1}) = \varphi(m_{i+1}, C_\iota) = 1/t \cdot (10tk^{i+1})^2 = \Omega(k^{2i+2})$. This term dominates $\Delta(b) = O((tk^i)^2) = O(k^{2i+0.2})$ where we used that all points affected by b have total weight of $O(1)$ and distance from b is at most $10tk^i$. So, we get $\Delta(m_i) > \Delta(b)$, as needed.

Next, consider any point $n_{i'}$ for $i' < i$; we will prove that $\Delta(n_{i'}) < \Delta(c)$. We have $\varphi(x, n_{i'}) < \varphi(x, c)$ only for $x = n_{i'}$ and $x = m_{i'}$. Using triangle inequality to bound $d(m_{i'}, c) \leq d(m_{i'}, n_{i'}) + d(n_{i'}, b) + d(b, c) = (10t + 1)k^j + 1$, we get

$$\varphi(m_{i'}, c) - \varphi(m_{i'}, n_{i'}) \leq 1000/t \cdot \left(((10t + 1)k^j + 1)^2 - (10tk^j)^2\right) \leq 30000k^{2i'}$$

(cf. Eq. (117)) and

$$\varphi(n_{i'}, c) - \varphi(n_{i'}, n_{i'}) = \varphi(n_{i'}, c) \leq 2000/t \cdot k^{2i'}$$

We will show that these terms are dominated by $\varphi(n_i, n_{i'}) - \varphi(n_i, c)$. First, using the cosine law, we have

$$\varphi(n_i, n_{i'}) - \varphi(n_i, b) = 1000/t \cdot \left(d(n_i, n_{i'})^2 - d(n_i, b)^2\right) \geq 1000/t \cdot 1/t \cdot d(b, n_{i'})d(b, n_i) = 1000k^{i+i'}/t^2$$

and

$$\varphi(n_i, c) - \varphi(n_i, b) = 1000/t \cdot (2d(b, c)d(b, n_i)/t + d(b, c)^2) \leq 1000k^i$$

Combining the two bounds, we get

$$\varphi(n_i, n_{i'}) - \varphi(n_i, c) \geq 1000k^{i+i'}/t^2 - 1000k^i$$

Since $i > i' \geq 1$, it is certainly true that $1000k^{i+i'}/t^2 - 1000k^i > 30000k^{2i'} + 2000k^{2i'}/t$ and we get that $\Delta(n_{i'}) < \Delta(c)$, as needed.

A very similar argument works for every $m_{i'}$ with $i' < i$ and we omit the proof.

The only missing point is now m_i for which we prove that $\Delta(m_i) < \Delta(c)$. To see that, note that the only point x for which $\varphi(x, m_i) < \varphi(x, c)$ is $x = m_i$ itself. We have

$$\varphi(m_i, b) - \varphi(m_i, m_i) = \varphi(m_i, b) = 1000/t \cdot ((1 + 10t)k^i)^2$$

On the other hand, we have

$$\varphi(b, m_i) - \varphi(b, b) = \varphi(b, m_i) = 1 \cdot ((1 + 10t)k^i)^2$$

That is, the cost drop of b if we take b dominates the cost drop of m_i if we take m_i. Whence $\Delta(m_i) < \Delta(b)$, as needed. $\qquad\square$

Consider the first $k^{0.5} - k^{0.4}$ steps of the i-th phase. We will show that what happens with high probability is that the algorithm select only points of $E_i \cup \{m_{i+1}\}$ as new centers, until at some point it selects n_i or b.

Claim 6.5. *Fix ι_0 to be the first step of phase i. Let ι_1 be the first point in time when either there were at least $k^{0.5} - k^{0.4}$ sampling steps of the i-th phase, or until $\{n_i, b\} \cap C_{\iota_1} \neq \emptyset$.*

Then, with probability at least $1 - 1/(10000t)$, we have $C_{\iota_1} \setminus C_{\iota_0} \subseteq E_i \cup \{n_i, b, m_{i+1}\}$ and either $\{n_i, m_{i+1}\} \subseteq C_{\iota_1}$ or $\{b\} \subseteq C_{\iota_1}$. Moreover, $b \in C_{\iota_1}$ with probability at least $1/(10^7 t)$.

Proof. We will need to upper bound the probability of various bad events. First, we upper bound the probability of the events that at least one point from $E_1 \cup \cdots \cup E_{i-1} \cup A$ is sampled as a candidate center. To compute the relevant probabilities, first note that at any point in time $\iota \leq \iota_1$, we have $|E_i| \geq k^{0.4}$, hence

$$\varphi(E_i, C_\iota) \geq k^{0.4} w_i \geq k^{0.4} \cdot k^{2i+2} \tag{119}$$

where we used Eq. (116).

On the other hand, we have $\varphi(E_1 \cup \cdots \cup E_{i-1}, C_\iota) \leq t\sqrt{k} w_{i-1} \leq 3000t\sqrt{k}k^{2i}$ using Eq. (116) and $\varphi(A, C_\iota) \leq k^{1.2} \cdot \frac{\log k}{k^2} \cdot (2k^{i+1})^2$. We get

$$\frac{\varphi(E_1 \cup \cdots \cup E_{i-1} \cup A, C_\iota)}{\varphi(X, C_\iota)} = O(\frac{k^{2i+1.3}}{k^{2i+2.4}}) = O(1/k).$$

Hence, during at most $k^{0.5} - k^{0.4} = O(\sqrt{k})$ steps of the phase, the probability of this event is at most $O(\sqrt{k} \cdot \ell/k) = O(1/k^{0.4})$.

Another bad event is that in some step of the algorithm none of the $\ell > 1$ points sampled is from E_i. To compute the probability of this event, we upper bound the following cost:

$$\begin{aligned}
&\varphi(\{b, c\} \cup N_{\leq i} \cup M_{\leq i+1} \cup E_1 \cup \cdots \cup E_{i-1} \cup A, C_\iota) \\
&= O(\varphi(m_{i+1}, n_{i+1})) \qquad\qquad\qquad\qquad \text{cost dominated by the cost of } m_{i+1} \\
&= O((10tk^{i+1})^2 \cdot 1/t) \\
&= O(tk^{2i+2})
\end{aligned}$$

On the other hand, in view of Eq. (119), we have

$$P(c_{\iota+1}^1, c_{\iota+1}^2 \notin E_i) = \left(\frac{O(tk^{2i+2})}{k^{2i+2.4}}\right)^2 = O(1/k^{0.7})$$

Hence, the probability that this bad event happens in the first $k^{0.5} - k^{0.4}$ steps is at most $O(k^{0.5}/k^{0.7}) = 1/k^{0.2}$.

A final bad event that we need to deal with is that in one sampling step we sample at least two points from the set $\{b, n_i, m_{i+1}\}$. We argue just about the probability that n_i, m_{i+1} are sampled in one sampling step as this event has the largest probability out of the three pairs $\{b, n_i\}, \{b, m_{i+1}\}, \{n_i, m_{i+1}\}$. The probability of this event in one step is at most

$$\frac{\ell\varphi(n_i, C_\iota)}{\varphi(X, C_\iota)} \cdot \frac{\ell\varphi(m_{i+1}, C_\iota)}{\varphi(X, C_\iota)} \leq \frac{\ell \cdot \frac{1000}{t} \cdot (2k^{i+1})^2 \cdot \ell \cdot \frac{1000}{t} \cdot (10tk^{i+1})^2}{(\gamma \cdot k^{2i+2})^2} = O((\ell/k^{0.4})^2) = O(1/k^{0.6})$$

where we used Eq. (119).

Hence, the probability that this bad event happens in the first $k^{0.5} - k^{0.4}$ steps is at most $O(k^{0.5}/k^{0.6}) = 1/k^{0.1}$.

In view of the computations above, we may assume that in the first $k^{0.5} - k^{0.4}$ steps we always sample one point from E_i and $\ell - 1$ points from $E_i \cup \{b, c\} \cup N_{\leq i} \cup M_{\leq i+1}$. Moreover, we do not sample more than one point from $\{b, n_i, m_{i+1}\}$ in one sampling step. We now invoke Claim 6.4 and get that only when the non-E_i point we sample is n_i or m_{i+1} or b, we add that point to the set of centers. Otherwise, we always select the new center as one of the sampled points from E_i.

We will now prove the claims from the statement. First, we prove that with probability $1 - O(1/t)$ we have either both n_i and m_{i+1} in C_{ι_1}, or we have $b \in C_{\iota_1}$.

In each step ι, there are at least $\gamma = k^{0.5} - (\iota - \iota_0) - 3$ not-yet-taken points of E_i, since we already assume that the algorithm chooses a point from $E_i \cup \{b, n_i, m_{i+1}\}$ as the next center in each step. The probability that the next point sampled is n_i is at least

$$\mathrm{P}(c_{\iota+1} = n_i) \geq \frac{\ell\varphi(n_i, C_\iota)}{\varphi(X, C_\iota)}$$
$$\geq \frac{\ell \cdot 1000/t \cdot k^{2i+2}}{10000k^{2i+2} + 3000\gamma k^{2i+2}} \geq \frac{\ell}{10t\gamma}$$

where we bounded $\varphi(\{b, c\} \cup A \cup N_{\leq i} \cup M_{\leq i+1}) \leq 10000k^{2i+2}$, used Eq. (116) and used that since $\gamma \geq k^{0.4}$, it dominates the constant term 10000. Hence, the probability that we do not take n_i nor b in the $k^{0.5} - k^{0.4}$ steps, is at most

$$\prod_{\gamma=k^{0.5}}^{k^{0.4}+3} (1 - \ell/(10t\gamma)) \leq e^{-\sum_{\gamma=k^{0.5}}^{k^{0.4}+3} \ell/(10t\gamma)} \leq e^{-(\ell\log k)/100t} = e^{-\log(\ell\log k)} < 1/(100000t)$$

where we used $t = \frac{\ell\log k}{1000\log(\ell\log k)}$ and summed up a harmonic series. Similarly, we can bound that the probability that we take neither m_{i+1} nor b is at most $1/(1000\ell\log k)$. The first part of the claim follows after we also subtract the probabilities of various bad events bounded above from $1 - 1/(100000t)$.

Note that after n_i is selected as a center during the phase in some step ι, the cost drop resulted by adding a point from $\{b, c\} \cup A \cup N_{<i} \cup M_{\leq i}$ to C_ι is smaller than the cost drop resulted by adding a point from E_i to C_ι. Hence, we have $C_{\iota_1} \setminus C_\iota \subseteq E_i \cup \{m_{i+1}\}$ which consequently implies $C_{\iota_1} \setminus C_{\iota_0} \subseteq E_i \cup \{n_i, b, m_{i+1}\}$ as required.

Finally, we need to prove that we sample b with probability at least $1/(100t)$. To see this, let us condition on b or n_i being one of the sampled and taken points in the first $n^{0.5} - n^{0.4}$ steps of the phase. In every step the ratio of the probability we sample b versus that we sample n_i is equal to

45

$$\varphi(b, C_\iota)/\varphi(n_i, C_\iota) \geq \left(\frac{\log^2(\ell \log k)}{1000 \ell^2 \log^2 k} \cdot k^{2i+2} \right) / \left(1000/t \cdot 2k^{2i+2} \right)$$

$$= \frac{\frac{\ell \log k}{\log(\ell \log k)} \cdot \log^2(\ell \log k)}{2 \cdot 10^6 \cdot \ell^2 \log^2 k}$$

$$= \frac{\log(\ell \log k)}{2 \cdot 10^6 \cdot \ell \log k} = 1/(2 \cdot 10^6 \cdot t)$$

Hence, after we subtract the probabilities of various bad events from $1/(2 \cdot 10^6 \cdot t)$, we get that the probability that we sample b during the process is at least $1/(10^7 t)$, as needed.

$\square$

We are now ready to deduce Claim 6.2.

Proof. We iterate Claim 6.5. As we have only $1/(10000t)$ probability of various failure modes inside one phase, the total probability of failing in some phase is at most $1/10000$. If we condition on no bad events happening in a phase, we get probability of at least $1/(10^7 t)$ of sampling b per phase. Hence, the total probability of sampling b in at least one phase is at least

$$1 - (1 - 1/10^7 t)^t \geq 1 - e^{-1/10^7} > 0$$

Finally, after b is sampled, we use Claim 6.3 to conclude that with positive probability, at the end of the first epoch we have $C_{\widetilde{k} - k^{1.2}} = X \setminus (A \cup \{c\})$ as needed. $\square$

It remains to argue about the second epoch. We need to prove that there is constant probability of sampling c as a candidate center during the first $k^{1.2} - k^{1.1}$ steps. Then we need to argue that in that case c is taken as a center by the greedy rule. We start by lower bounding the probability of sampling c.

Claim 6.6. *Assume that in some step ι_0 we have $X \setminus C_{\iota_0} \subseteq \{c\} \cup A$ but $c \notin C_{\iota_0}$. Moreover, assume that $k^{1.1} \leq |A|$. Then, with probability at least $1 - e^{-1/(6 \log k)}$, Algorithm 2 samples the point c as a candidate center in the following $|A|/2$ steps.*

Proof. In each of the next $|A|/2$ steps $\iota_0 \leq \iota \leq \iota_0 + |A|/2$, unless $c \in C_\iota$, we sample c as a fixed candidate center c_ι^j with probability at least $\frac{\varphi(c,b)}{\varphi(c,b) + |A| \cdot \varphi(a_1, b)} \geq \frac{1 \cdot 1^2}{1 \cdot 1^2 + |A| \cdot \frac{\ell \log k}{k^2} \cdot (k+1)^2} \geq \frac{1}{3|A|\ell \log k}$.

Hence, the probability that c is not sampled in any of the $|A|/2$ steps as no candidate center c_ι^j is at most

$$\left(1 - \frac{1}{3|A|\ell \log k} \right)^{\ell \cdot |A|/2} \leq e^{-\frac{1}{3|A|\ell \log k} \cdot \ell |A|/2} = e^{-1/(6 \log k)}$$

and we are done. $\square$

We can now finish the proof.

Claim 6.7. *Assume that $C_{\widetilde{k} - k^{1.2}} = X \setminus (A \cup \{c\})$. Then, with positive probability we have $c \in C_{\widetilde{k}}$.*

46

Proof. Consider the first $k^{1.2} - k^{1.1}$ steps after the end of the first epoch. We split these steps to $\log_2 \frac{k^{1.2}}{k^{1.1}} = 0.1 \log_2 k$ batches where the batch i contains $k^{1.2}/2^i$ steps. Fix one such i-th batch. We first prove that whenever the algorithm samples a point c as a candidate center, it also takes it as a center. To see that, we need to compute the drop in cost $\Delta(c)$ of c and $\Delta(a_j)$ of any a_j. First, we bound the drop of c. We use the fact that each point of at least $k^{1.1}$ points A that are not yet taken will have its cost dropped by a small yet non-negligible amount. Namely for the drop in the cost $\Delta(c)$ after taking c as a new center we have

$$\Delta(c) \geq \sum_{j=1}^{k^{1.1}} \varphi(a_j, b) - \varphi(a_j, c) \geq k^{1.1} \cdot \frac{\ell \log k}{k^2} \cdot \left(d(b, a_1)^2 - d(c, a_1)^2 \right) \tag{120}$$

$$\geq \frac{\ell \log k}{k^{0.9}} \cdot k/2 \qquad \text{Eq. (108)} \tag{121}$$

$$\geq \ell k^{0.1} \tag{122}$$

On the other hand, whenever we take some point a_j as a center, the only point whose cost is dropped is a_j itself; this follows from Eq. (107). Thus for the drop in the cost $\Delta(a_j)$ after taking a_j as a new center we have

$$\Delta(a_j) = \varphi(a_j, b) = \frac{\ell \log k}{k^2} \cdot k^2 = \ell \log k \tag{123}$$

That is, for k large enough we get $\Delta(c) > \Delta(a_j)$ and, hence, whenever c is sampled, it is also taken as a center by the greedy rule of Algorithm 2.

Finally, we use Claim 6.6 to conclude that c is not sampled in any of the $0.1 \log_2 k$ batches only with probability at most $\left(e^{-1/(6 \log k)} \right)^{0.1 \log_2 k} \leq e^{-1/100} < 1$. □

References

[ACKS15] Pranjal Awasthi, Moses Charikar, Ravishankar Krishnaswamy, and Ali Kemal Sinop. The hardness of approximation of euclidean k-means. *arXiv preprint arXiv:1502.03316*, 2015.

[ADHP09] Daniel Aloise, Amit Deshpande, Pierre Hansen, and Preyas Popat. Np-hardness of euclidean sum-of-squares clustering. *Machine learning*, 75(2):245–248, 2009.

[ADK09] Ankit Aggarwal, Amit Deshpande, and Ravi Kannan. Adaptive sampling for k-means clustering. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 15–28. Springer, 2009.

[ANFSW19] Sara Ahmadian, Ashkan Norouzi-Fard, Ola Svensson, and Justin Ward. Better guarantees for k-means and euclidean k-median by primal-dual algorithms. *SIAM Journal on Computing*, (0):FOCS17–97, 2019.

[AV07]   David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seed-
            ing. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete
            algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.

[BERS20]   Anup Bhattacharya, Jan Eube, Heiko Röglin, and Melanie Schmidt. Noisy, greedy
            and not so greedy k-means++. In *28th Annual European Symposium on Algorithms
            (ESA 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[BLHK16a]   Olivier Bachem, Mario Lucic, Hamed Hassani, and Andreas Krause. Fast and provably
            good seedings for k-means. In *Advances in neural information processing systems*,
            pages 55–63, 2016.

[BLHK16b]   Olivier Bachem, Mario Lucic, S Hamed Hassani, and Andreas Krause. Approximate k-
            means++ in sublinear time. In *Thirtieth AAAI Conference on Artificial Intelligence*,
            2016.

[BLK17]   Olivier Bachem, Mario Lucic, and Andreas Krause. Distributed and provably good
            seedings for k-means in constant rounds. In *Proceedings of the 34th International
            Conference on Machine Learning-Volume 70*, pages 292–300. JMLR. org, 2017.

[BMV⁺12]   Bahman Bahmani, Benjamin Moseley, Andrea Vattani, Ravi Kumar, and Sergei Vas-
            silvitskii. Scalable k-means++. *Proceedings of the VLDB Endowment*, 5(7):622–633,
            2012.

[BR13]   Tobias Brunsch and Heiko Röglin. A bad instance for k-means++. *Theoretical Com-
            puter Science*, 505:19–26, 2013.

[BVX19]   Aditya Bhaskara, Sharvaree Vadgama, and Hong Xu. Greedy sampling for approxi-
            mate clustering in the presence of outliers. *Advances in Neural Information Processing
            Systems*, 32, 2019.

[CAEMN22]   Vincent Cohen-Addad, Hossein Esfandiari, Vahab Mirrokni, and Shyam Narayanan.
            Improved approximations for euclidean k-means and k-median, via nested quasi-
            independent sets, 2022.

[CGPR20]   Davin Choo, Christoph Grunau, Julian Portmann, and Václav Rozhoň. k-means++:
            few more steps yield constant approximation, 2020.

[CKV13]   M Emre Celebi, Hassan A Kingravi, and Patricio A Vela. A comparative study of
            efficient initialization methods for the k-means clustering algorithm. *Expert systems
            with applications*, 40(1):200–210, 2013.

[Das19]   Sanjoy Dasgupta. Lecture 3 – algorithms for k-means clustering, 2013. accessed May
            8th, 2019.

[FMS07]   Dan Feldman, Morteza Monemizadeh, and Christian Sohler. A ptas for k-means clus-
            tering based on weak coresets. In *Proceedings of the twenty-third annual symposium
            on Computational geometry*, pages 11–18, 2007.

[GOR]   Christoph Grunau, Ahmet Ozudogru, and Vaclav Rozhon. Noisy k-means++ revis-
            ited.

[GR20] Christoph Grunau and Václav Rozhoň. Adapting k-means algorithms for outliers, 2020.

[KMN+04] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. A local search approximation algorithm for k-means clustering. *Computational Geometry*, 28(2-3):89–112, 2004.

[KSS04] Amit Kumar, Yogish Sabharwal, and Sandeep Sen. A simple linear time (1+/spl epsiv/)-approximation algorithm for k-means clustering in any dimensions. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 454–462. IEEE, 2004.

[Llo82] Stuart P. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–136, 1982.

[LS19] Silvio Lattanzi and Christian Sohler. A better k-means++ algorithm via local search. In *International Conference on Machine Learning*, pages 3662–3671, 2019.

[LSW17] Euiwoong Lee, Melanie Schmidt, and John Wright. Improved and simplified inapproximability for k-means. *Information Processing Letters*, 120:40–43, 2017.

[MNV09] Meena Mahajan, Prajakta Nimbhorkar, and Kasturi Varadarajan. The planar k-means problem is np-hard. In *International Workshop on Algorithms and Computation*, pages 274–285. Springer, 2009.

[MRS20] Konstantin Makarychev, Aravind Reddy, and Liren Shan. Improved guarantees for k-means++ and k-means++ parallel. *Advances in Neural Information Processing Systems*, 33:16142–16152, 2020.

[MV20] Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions, 2020.

[ORSS13] Rafail Ostrovsky, Yuval Rabani, Leonard J Schulman, and Chaitanya Swamy. The effectiveness of lloyd-type methods for the k-means problem. *Journal of the ACM (JACM)*, 59(6):1–22, 2013.

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[R C13] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013.

[Roz20] Václav Rozhoň. Simple and sharp analysis of k-means——, 2020.

[Vas07] Sergei Vassilvitskii. k-means: algorithms, analyses, experiments, 2007.

[Wei16] Dennis Wei. A constant-factor bi-criteria approximation guarantee for k-means++. In *Advances in Neural Information Processing Systems*, pages 604–612, 2016.

Algorithm 3 General k-means++ seeding

Input: X, k, ℓ, and a rule $\mathcal{R}$ that picks one point from ℓ points $x_1, \ldots, x_\ell$, given access to X, C_i, k, ℓ.

1: Uniformly independently sample $x_1, \ldots, x_\ell \in X$;
2: Let $x \in \{x_1, \ldots, x_\ell\}$ be selected by $\mathcal{R}$ and set $C_1 = \{x\}$.
3: **for** $i \leftarrow 1, 2, 3, \ldots, k - 1$ **do**
4:      Sample $c_{i+1}^1, \ldots, c_{i+1}^\ell \in X$ independently w.p. $\frac{\varphi(x, C_i)}{\varphi(X, C_i)}$;
5:      Let $c_{i+1} \in \{c_{i+1}^1, \ldots, c_{i+1}^\ell\}$ be selected by $\mathcal{R}$ and set $C_{i+1} = C_i \cup \{c_{i+1}\}$.
6: **return** $C := C_k$

A   A hard instance for general k-means++

In this section, we prove the precise version of Theorem 1.3. To this end, we first formally describe the general version of k-means++ with for an arbitrary seeding rule in Algorithm 3.

The rest of this section is then devoted to the proof of the following theorem.

Theorem A.1. *There exists a point set $X \subseteq \mathbb{R}^d$ and a rule $\mathcal{R}$ such that Algorithm 3 with $\mathcal{R}$ is $\Omega(k^{1-1/\ell})$-approximate with constant probability.*

We already sketched a proof of this theorem in Section 2.2 for $\ell = \Omega(\log k)$. The generalization for any ℓ and to a Euclidean space below is routine.

We remark that we believe one can improve the lower bound from Theorem A.1 to $\Omega(k^{1-1/\ell}\ell \log k)$ by adding the set A as in the proof of Theorem 1.2.

We begin by describing the input instance (X, k, C_0) (recall that in view of Lemma 3.4 we can assume we start with a non-empty set of centers C_0). Throughout the proof, we will assume that the weights are integers. This is for readability, we could round the numbers to the closest integer and that would not hurt any asymptotic guarantees. Our instance X is a subset of $k+1$-dimensional Euclidean space R^{k+1}.

We next describe the input weighted point set X.

1. There is a point $d \in C_0$ in the origin.

2. There are $k - 1$ points $x_1, x_2, \ldots, x_{k-1}$ having weight $w(x_i) = 1$. Moreover each x_i has the coordinate $(0, \ldots, 0, k, 0, \ldots, 0)$, which has value 0 at each dimension except the i-th. Hence, $d(d, x_i) = k$ for every $i \in \{1, 2, \ldots, k-1\}$.

3. There is a point c with weight $w(c) = \frac{k^{1-1/\ell}}{2}$ at $(0, 0, \ldots, k, 0)$. Hence $d(d, c) = k$.

4. The final point b has weight $w(b) = 1$ and lies in the plane generated by vectors $(0, \ldots, 1, 0)$ and $(0, \ldots, 0, 1)$ in such a position that it holds that $d(c, d) = d(b, d) = k$ and $d(b, c) = 1$ (i.e., d, c, b form an isosceles triangle).

In view of Lemma 3.5, we require the optimal solution $C^* \subseteq X$. Then, we have $C^* = \{x_1, x_2, \ldots, x_{k-1}, c\}$ and the cost of it is $OPT = \varphi(b, C^* \cup C_0) = d(c, b)^2 \cdot w(b) = 1$.

We are going to pick the rule $\mathcal{R}$ as follows: whenever we sample b as a candidate, we take it as a center. Furthermore, we only take c as a center if all of the ℓ candidate points are c, hence when we have no other choice.

We will show that with constant probability we will take b as a center after $k/2$ steps. That means that at least one of the points in $\{x_1, x_2, \ldots, x_{k-1}, c\}$ will not be selected as a center at the end. If one of the x_i is not selected as a center, then the cost of the solution will be at least

$w(x_i) \cdot d(x_i, d)^2 = k^2$, since d is the closest point to any x_i. If c is not selected as a center, then the cost of the solution will be at least $w(c) \cdot d(c, b)^2 = \frac{k^{1-1/\ell}}{2}$, since the closest chosen center to c is b. Hence if we pick b as a center, the approximation factor will be at least $\frac{k^{1-1/\ell}}{2}$.

Let B_i and C_i be the events that b and c are chosen as a center at the i-th step, respectively and $B_{\leq i}$ and $C_{\leq i}$ be the events that b and c are chosen as a center in one of the first i steps.

We will calculate the probability of picking b as a center in the first $k/2$ steps as follows:

$$P(B_{\leq k/2}) = P(B_1) + P(B_2 \mid \neg B_{\leq 1}) \cdot P(\neg B_{\leq 1}) + \cdots + P(B_{k/2} \mid \neg B_{\leq k/2-1}) \cdot P(\neg B_{\leq k/2-1})$$

To calculate a lower bound for $P(B_{\leq k/2})$, we will need a lower bound for $P(B_i \mid \neg B_{\leq i-1})$ and $P(\neg B_{\leq i})$. We start by showing that with constant probability we do not pick b in the first $k/2$ steps.

Lemma A.2. *For any $i \leq k/2$ we have $P(\neg B_{\leq i}) \geq \frac{1}{10}$.*

Proof. First we show that for any $i \leq \frac{k}{2}$ we have $P(\neg B_i | \neg B_{\leq i-1}) \geq 1 - \frac{2}{k+2}$. We have

$$
\begin{aligned}
P(\neg B_i \mid \neg B_{\leq i-1} \wedge \neg C_{\leq i-1}) &= \frac{k - i + k^{1-1/\ell}/2}{k - i + k^{1-1/\ell}/2 + 1} \\
&\geq \frac{k/2 + k^{1-1/\ell}/2}{k/2 + k^{1-1/\ell}/2 + 1} \\
&\geq \frac{k/2}{k/2 + 1} = 1 - \frac{k/2}{k/2 + 1} = 1 - \frac{2}{k+2}
\end{aligned}
$$

$$
\begin{aligned}
P(\neg B_i \mid \neg B_{\leq i-1} \wedge C_{\leq i-1}) &= \frac{(k - i + 1) \cdot k^2}{1 + (k - i + 1) \cdot k^2} \\
&\geq \frac{k^3/2}{1 + k^3/2} = 1 - \frac{2}{2 + k^3} \geq 1 - \frac{2}{k+2}
\end{aligned}
$$

Hence $P(\neg B_i \mid \neg B_{\leq i-1}) \geq 1 - \frac{2}{k+2}$. Now we can calculate the probability of not picking b in the first i steps.

$$
\begin{aligned}
P(\neg B_{\leq i}) &= P(\neg B_1 \cap \neg B_2 \cap \neg B_3 \cap \cdots \cap \neg B_i) \\
&= P(\neg B_1 \mid \neg B_{\leq 0}) \cdot P(\neg B_2 \mid \neg B_{\leq 1}) \cdot P(\neg B_3 \mid \neg B_{\leq 2}) \cdots P(\neg B_i \mid \neg B_{\leq i-1}) \\
&\geq \left(1 - \frac{2}{k+2}\right)^i \\
&\geq \left(1 - \frac{2}{k+2}\right)^{k/2} \\
&\geq \left(e^{-\frac{4}{k+2}}\right)^{k/2} \qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{\textcolor{red}{Fact 3.8}} \\
&\geq \left(e^{-\frac{2k}{k+2}}\right) \geq \frac{1}{10}
\end{aligned}
$$

$\square$

Now, to calculate a lower bound for $P(B_i \mid \neg B_{\leq i-1})$ we split it as follows:

$$P(B_i \mid \neg B_{\leq i-1}) = P(B_i \mid \neg B_{\leq i-1} \wedge \neg C_{\leq i-1}) \cdot P(\neg C_{\leq i-1}) + P(B_i \mid \neg B_{\leq i-1} \wedge C_{\leq i-1}) \cdot P(C_{\leq i-1})$$
$$\geq P(B_i \mid \neg B_{\leq i-1} \wedge \neg C_{\leq i-1}) \cdot P(\neg C_{\leq i-1})$$

First, we will show that the probability of picking c in a step is at most $1/k$.

Lemma A.3. *For every $i \leq k/2$ we have $P(C_i \mid \neg C_{\leq i-1}) \leq 1/k$.*

Proof. We will start by showing that the probability of c being selected as a center is higher when b is not selected as a center. This intuitively makes sense because b lies very close to c.

$$P(C_i \mid \neg C_{\leq i-1} \wedge B_{\leq i-1}) = \frac{k^{1-1/\ell}}{k^{1-1/\ell} + (k-i+1) \cdot k^2}$$
$$\leq \frac{k^{1-1/\ell} + (k^{3-1/\ell} - k^{1-1/\ell})}{k^{1-1/\ell} + (k-i+1) \cdot k^2 + (k^{3-1/\ell} - k^{1-1/\ell})}$$
$$\leq \frac{k^{3-1/\ell}}{k^{3-1/\ell} + (k-i+1) \cdot k^2} = P(C_i \mid \neg C_{\leq i-1} \wedge \neg B_{\leq i-1})$$

Now using $P(C_i \mid \neg C_{\leq i-1} \wedge B_{\leq i-1}) \leq P(C_i \mid \neg C_{\leq i-1} \wedge \neg B_{\leq i-1})$, we can bound $P(C_i \mid \neg C_{\leq i-1})$.

$$P(C_i \mid \neg C_{\leq i-1}) = P(B_{\leq i-1}) \cdot P(C_i \mid \neg C_{\leq i-1} \wedge B_{\leq i-1}) + P(\neg B_{\leq i-1}) \cdot P(C_i \mid \neg C_{\leq i-1} \wedge \neg B_{\leq i-1})$$
$$\leq P(B_{\leq i-1}) \cdot P(C_i \mid \neg C_{\leq i-1} \wedge \neg B_{\leq i-1}) + P(\neg B_{\leq i-1}) \cdot P(C_i \mid \neg C_{\leq i-1} \wedge \neg B_{\leq i-1})$$
$$\leq P(C_i \mid \neg C_{\leq i-1} \wedge \neg B_{\leq i-1})$$

According to our rule $\mathcal{R}$, for c to be selected as a center, all of the ℓ samples in a step should be c.

$$P(C_i \mid \neg C_{\leq i-1} \wedge \neg B_{\leq i-1}) = \left(\frac{k^{3-1/\ell}/2}{k^{3-1/\ell}/2 + (k-i+1) \cdot k^2} \right)^\ell$$
$$\leq \left(\frac{k^{3-1/\ell}/2}{k^{3-1/\ell}/2 + k/2 \cdot k^2} \right)^\ell$$
$$\leq \left(\frac{k^{3-1/\ell}/2}{k^3/2} \right)^\ell$$
$$\leq \left(\frac{1}{k^{1/\ell}} \right)^\ell = \frac{1}{k}$$

Hence $P(C_i \mid \neg C_{\leq i-1}) \leq P(C_i \mid \neg C_{\leq i-1} \wedge \neg B_{\leq i-1}) \leq \frac{1}{k}$ □

Now we can show that the probability of not picking c as a center in the first $k/2$ steps is at least $1/2$.

Lemma A.4. *For $i \leq k/2$, $P(\neg C_{\leq i}) \geq \frac{1}{2}$*

Proof.

$$\begin{aligned}
\mathrm{P}(C_{\leq i}) &= \mathrm{P}(C_1) + \mathrm{P}(C_2 \mid \neg C_{\leq 1}) \cdot \mathrm{P}(\neg C_{\leq 1}) + \cdots + \mathrm{P}(C_i \mid \neg C_{\leq i-1}) \cdot \mathrm{P}(\neg C_{\leq i-1}) \\
&\leq \mathrm{P}(C_1 \mid \neg C_{\leq 0}) + \mathrm{P}(C_2 \mid \neg C_{\leq 1}) + \mathrm{P}(C_3 \mid \neg C_{\leq 2}) + \cdots + \mathrm{P}(C_i \mid \neg C_{i-1}) \\
&\leq \frac{1}{k} + \frac{1}{k} + \frac{1}{k} + \cdots + \frac{1}{k} && \text{\textcolor{red}{Lemma A.3}} \\
&\leq \frac{i}{k} \leq \frac{1}{2}
\end{aligned}$$

Hence, $\mathrm{P}(\neg C_{\leq i}) = 1 - \mathrm{P}(C_{\leq i}) \geq \frac{1}{2}$ $\qquad\square$

Using Lemma A.4 we can finally calculate a lower bound for $\mathrm{P}(B_i \mid \neg B_{\leq i-1})$.

Lemma A.5. *For $i \leq k/2$, $P(B_i \mid \neg B_{\leq i-1}) \geq \frac{1}{4k}$*

Proof.

$$\begin{aligned}
\mathrm{P}(B_i \mid \neg B_{\leq i-1}) &= \mathrm{P}(B_i \mid \neg B_{\leq i-1} \wedge \neg C_{\leq i-1}) \cdot \mathrm{P}(\neg C_{\leq i-1}) + \mathrm{P}(B_i \mid \neg B_{\leq i-1} \wedge C_{\leq i-1}) \cdot \mathrm{P}(C_{\leq i-1}) \\
&\geq \mathrm{P}(B_i \mid \neg B_{\leq i-1} \wedge \neg C_{\leq i-1}) \cdot \mathrm{P}(\neg C_{\leq i-1}) \\
&\geq \mathrm{P}(B_i \mid \neg B_{\leq i-1} \wedge \neg C_{\leq i-1}) \cdot \frac{1}{2} && \text{\textcolor{red}{Lemma A.4}} \\
&\geq \frac{1}{(k-i) + k^{1-1/\ell}/2 + 1} \cdot \frac{1}{2} \\
&\geq \frac{1}{2k} \cdot \frac{1}{2} = \frac{1}{4k}
\end{aligned}$$

$\qquad\square$

Now we are ready to prove the theorem.

$$\begin{aligned}
\mathrm{P}(B_{\leq k/2}) &= \mathrm{P}(B_1) + \mathrm{P}(B_2 \mid \neg B_{\leq 1}) \cdot \mathrm{P}(\neg B_{\leq 1}) \cdots + \mathrm{P}(B_{k/2} \mid \neg B_{\leq k/2-1}) \cdot \mathrm{P}(\neg B_{\leq k/2-1}) \\
&\geq \frac{1}{10} \cdot \big(\mathrm{P}(B_1 \mid \neg B_{\leq 0}) + \mathrm{P}(B_2 \mid \neg B_{\leq 1}) + \mathrm{P}(B_3 \mid \neg B_{\leq 2}) + \cdots + \mathrm{P}(B_{k/2} \mid \neg B_{\leq k/2-1}) \big) && \text{\textcolor{red}{Lemma A.2}} \\
&\geq \frac{1}{10} \cdot (\frac{1}{4k} + \frac{1}{4k} + \frac{1}{4k} + \cdots + \frac{1}{4k}) && \text{\textcolor{red}{Lemma A.5}} \\
&\geq \frac{1}{10} \cdot (\frac{k/2}{4k}) \\
&\geq \frac{1}{80}
\end{aligned}$$

Hence, with constant probability, b will be taken as a center in the first $k/2$ steps, and as a result, the algorithm will return a solution with an approximation ratio of at least $\Omega(k^{1-1/\ell})$.

B   Analysis of general k-means++

In this section, we prove the precise version of Theorem 1.4 that we state next.

Theorem B.1. *For any rule $\mathcal{R}$, Algorithm 3 is $O(k^{2-1/\ell} \cdot \ell \log k)$-approximate.*

B.1 Hitting optimal clusters

We start by proving an analogue to the Lemma 2.2 that shows that any optimal cluster is expected to be hit $O(\ell k^{1-1/\ell})$ times. Note that it is trivially hit $O(\ell k)$ times. The reason we bother proving this only slightly better (and tight) result is that we wrote the proof before we realized that our lower and upper bounds for Algorithm 3 are not matching since we could not analyze the sampling process defined below.

The improvement over the trivial $O(\ell k)$ bound is based on the fact that if a cluster K dominates the cost of the whole point set, we have a nontrivial probability of sampling all ℓ candidate centers from it.

Lemma B.2. *For any rule $\mathcal{R}$ in Algorithm 3 and any optimal cluster K we have that $E[HIT(K)] = O(\ell \cdot k^{1-1/\ell})$.*

Proof. We prove this statement by induction. Recall that $\mathrm{HIT}(K)$ be the number of points of K that we sample from K until K becomes covered (Definition 4.1) or solved (Definition 4.2) but for the purposes of this proof we even drop the "solved" requirement.

We prove that

$$\mathrm{E}_{\geq k-i}[\mathrm{HIT}_{\geq k-i}(K)] \leq 10\ell i^{1-1/\ell}$$

For $i = 0$ it clearly holds. Next, assume the equation holds for $k - i + 1$ and we prove it for $k - i$.

Let us define $p = \frac{\varphi(K,C_i)}{\varphi(X,C_i)}$. We will now use the fact that whenever we sample all points $c_i^1, \ldots, c_i^\ell$ from K, i.e., whenever $\mathrm{HIT}_i(K) = \ell$, we have $c_i \in K$ and K hence becomes covered. Namely, using induction hypothesis we compute:

$$\mathrm{E}_{\geq k-i}[\mathrm{HIT}_{\geq k-i}(K)] = \mathrm{E}_{k-i}[\mathrm{HIT}_i(K)] + \mathrm{E}_{\geq k-i}[\mathrm{HIT}_{\geq k-i+1}(K)) \tag{124}$$

$$\leq \ell p + \mathrm{P}(\mathrm{HIT}_i(K) = \ell) \cdot 0 + \mathrm{P}(\mathrm{HIT}_i(K) \neq \ell)10\ell(i-1)^{1-1/\ell} \tag{125}$$

$$= \ell p + (1 - p^\ell)10\ell(i-1)^{1-1/\ell} \tag{126}$$

Next, we compute

$$(i-1)^{1-1/\ell} = i^{1-1/\ell} \cdot (1-1/i)^{1-1/\ell}$$
$$\leq i^{1-1/\ell} \cdot (1-1/i)^{1/2} \qquad\qquad \ell \geq 2$$
$$\leq i^{1-1/\ell} \cdot (1-1/(4i))$$

Let us use $f(i) = 10\ell i^{1-1/\ell} \cdot (1-1/(4i))$. Then, the above computation in Eq. (124) says that

$$\mathrm{E}_{\geq k-i}[\mathrm{HIT}_{\geq k-i}(K)] \leq \ell p + (1-p^\ell)f(i) \tag{127}$$

Let us analyze the right-hand side of that expression. We have

$$\frac{\delta\left(\ell p + (1-p^\ell)f(i)\right)}{\delta p} = \ell - \ell p^{\ell-1}f(i)$$

Solving for the right hand side equal to zero, we get $1 - p^{\ell-1}f(i) = 0$, hence $p = (1/f(i))^{1/(\ell-1)}$. That is, the right hand side of Eq. (127) is maximized for that p and then it is equal to

$$\ell(1/f(i))^{1/(\ell-1)} + \left(1 - \left((1/f(i))^{1/(\ell-1)}\right)^\ell\right)f(i) \tag{128}$$

Let us plug in the definition of $f(i)$ to that expression. We start with the first term. We have

$$\ell\,(1/f(i))^{1/(\ell-1)} = \ell\left(\frac{1}{10\ell i^{1-1/\ell}\cdot(1-1/(4i))}\right)^{1/(\ell-1)} \leq \frac{\ell}{i^{1/\ell}}$$

where we used that $10\ell(1-1/(4i)) \geq 1$.

Next, we handle the second term as follows:

$$\left(1-\left((1/f(i))^{1/(\ell-1)}\right)^{\ell}\right)f(i) \leq f(i) \leq 10\ell i^{1-1/\ell}\cdot(1-1/(4i))$$

Hence, we can upper bound the expression in Eq. (128) by

$$10\ell i^{1-1/\ell}\cdot(1-1/(4i)) + \frac{\ell}{i^{1/\ell}}$$
$$= 10\ell i^{1-1/\ell}\cdot\left(1-1/(4i)+\frac{1}{10i}\right)$$
$$\leq 10\ell i^{1-1/\ell}$$

and we are done.

$\square$

As a corollary we get Theorem B.1.

Proof Sketch. The proof is very similar to the proof of Theorem 1.1. In that proof, we are using the greedy rule in two places:

1. Through Lemma 2.2; instead of that lemma we now use Lemma B.2.

2. Inside Proposition 5.6 we use it to bound how much can the size of the average uncovered cluster increases during the algorithm. We can very crudely bound this multiplicative increase by k, hence our approximation guarantee picks up additional k-factor.

$\square$

B.2 An interesting sampling process

This section is devoted to the explanation of an interesting open problem that, if solved, probably brings together the upper and lower bounds for Algorithm 3 that are now off by a factor of k. We note that losing a factor of k because of the drift of the size of the average uncovered cluster looks very wasteful. In fact, we can replace this factor of k in the upper bound by a function $g(k,\ell)$ that we discuss in the rest of this section. We believe that understanding $g(k,\ell)$ is an exciting open problem. The problem in the analysis of Theorem B.1 can be distilled into the following riddle, which one can understand without understanding the details of the analysis of Arthur and Vassilvitskii.

Definition B.3 (ℓ-point adversarial sampling process)**.** *Let $\ell \in \mathbb{N}$. We define the ℓ-point adversarial sampling process as follows. At the beginning, there is a set E_0 of k elements where each element $e \in E_0$ has some nonnegative weight $w_0(e)$. The process has k rounds: in each round, we form the new set E_{i+1} from E_i as follows:*

1. We define the distribution D_i over E_i where the probability of e is defined as $w_i(e)/\sum_{e\in E_i} w_i(e)$. An adversary chooses an arbitrary number ℓ_i that satisfies $0 \le \ell_i \le \ell$. We sample ℓ_i points $e_i^1, \ldots, e_i^{\ell_i}$ independently from D_i. Next, an adversary chooses a point $e_i \in \{e_i^1, \ldots, e_i^{\ell_i}\}$. We set $E_{i+1} = E_i \setminus \{e_i\}$.

2. An adversary chooses the new weight function $w_{i+1}(e)$ for every element $e \in E_{i+1}$ as an arbitrary function that satisfies
$$0 \le w_{i+1}(e) \le w_i(e).$$

The relationship between this process and the analysis of Algorithm 3 is as follows. The set E_i of elements corresponds to a set of uncovered clusters. The steps where we sample $\ell_i \le \ell$ elements $e_i^1, \ldots, e_i^{\ell_i}$ corresponds to the algorithm sampling at most ℓ centers from uncovered clusters. We assume that the rule $\mathcal{R}$ behaves adversarially and can decide to cover any of the sampled clusters, i.e., we allow removing any sampled element $e_i \in \{e_i^1, \ldots, e_i^{\ell_i}\}$ from E_i to form E_{i+1}. The adversarial decreasing of the element weights in between two sampling steps corresponds to the newly taken center decreasing the cost of the optimal clusters in an uncontrolled manner.

We note that the analysis of k-means++ from [AV07] implicitly analyzes this game for $\ell = 1$. This case is qualitatively simpler than the general case: We can in fact even prove that for $\ell = 1$, the average element size can only decrease between two steps. To see this, we observe that it would stay the same if we picked each element uniformly at random. Picking heavier elements with higher probability can only decrease the average size then.

However, this simple approach does not work anymore for $\ell > 1$. In fact, consider as an example the set E_0 consisting of $k-1$ elements of size one and one element of size k. Choosing $\ell = \Omega(k)$, the adversary can prevent us to remove the costly element until the very end, with constant probability. This increases the average element size from roughly 2 to k, that is, by $\Omega(k) = \Omega(\ell)$ factor.

We do not know how much the average can increase but it is clearly at most by a $O(k)$ factor and by the above reasoning it is at least by $\Omega(\ell)$ factor.

Fact B.4. *We define the function $g(k, \ell)$ as the smallest growing function satisfying the following condition. Let AVG_i be defined as the average weight of an element in the i-th round of the ℓ-point adversarial sampling process from Definition B.3, i.e., for any $0 \le i < k$ we define*
$$AVG_i = \frac{\sum_{e\in E_i} w_i(e)}{k - i}.$$
Then, for any adversary and any $0 \le i < k$, we have
$$AVG_i \le g(k, \ell)AVG_0.$$

Question B.5. *What is the value of $g(k, \ell)$?*

A similar problem to our sampling process was recently considered in [BERS20] where the authors consider the following related problem. Suppose that we run k-means++, but before each sampling step, an adversary distorts each probability of taking an element by a multiplicative $1 \pm \varepsilon$ factor. Does such an algorithm retain $O(\log k)$ approximation guarantees for fixed $\varepsilon < 1$? This question leads to the analysis of a process very similar to Definition B.3; instead of choosing one of ℓ sampled elements, the power of the adversary is now to distort the sampling distribution pointwise by $1 \pm \varepsilon$ multiplicative factor. In [BERS20], the authors show that the average in this game can increase only by a multiplicative $O(\log k)$ factor. This follows from the fact that all elements larger than $\Omega(\log k)$ will be already taken in the first $k/2$ steps of the process. This implies $O(\log^2 k)$ approximation guarantee for the final algorithm. In [GOR] this analysis of the game is improved to $O(1)$ which implies the tight $O(\log k)$ upper bound for k-means++ with noise.

C An incomparable bound on the number of hits

In this section, we sketch the proof of the following result which is incomparable with Lemma 2.2 but substantially easier to prove. In fact, we used this proof sketch as a way to build intuition towards the proof of Lemma 2.2 in an earlier draft of this writeup, before we realized Sections 1 and 2 are way too long.

Lemma C.1. *For any optimal cluster K we have $E[HIT(K)] = O(\ell \log \frac{OPT(1)}{OPT(K)})$.*

Here, $OPT(\widetilde{k})$ is the size of the optimal solution with $\widetilde{k}$ centers.

Proof sketch. Fix an optimal cluster K, consider a step $i+1$ and assume that for the cost of K we have $\varphi(K, C_i) \geq 10^5 \varphi^*(K)$. Being far from the optimum means that all centers of C_i are very far from most of the points of K. Hence, whenever it happens that a potential center c_{i+1}^j for some $1 \leq j \leq \ell$ is sampled from K, we have constant probability that $d(\mu(K), c_{i+1}^j) \leq d(\mu(K), C_i)/2$, i.e., c_{i+1}^j is substantially closer to $\mu(K)$ than all other centers in C_i. In that case, we have $\varphi(X, C_i) - \varphi(X, C_i \cup \{c_{i+1}^j\}) \geq \varphi(K, C_i)/2$. That is, adding c_{i+1}^j as the new center will result in the cost drop of at least $\varphi(K, C_i)/2$.

We will now need to distinguish two cases. Let us consider the distribution over the cost drop $\varphi(X, C_i) - \varphi(X, C_i \cup \{c\})$ where c is sampled proportional to its current cost $\varphi(c, C_i)$. That is, we consider the distribution of how the cost drops if we add the candidate center c_{i+1}^1 (or any other fixed candidate center) to the current solution. In the first, *easy*, case we assume that with probability $1 - 1/\ell$, the cost drop $\varphi(X, C_i) - \varphi(X, C_i \cup \{c\})$ is less than $\varphi(K, C_i)/2$; otherwise we are in the *hard* case.

What is easy in the easy case? The discussion above implies that in that case, whenever we sample some c_{i+1}^j from K, we have constant probability that all other candidate centers create a cost drop smaller than $\varphi(K, C_i)/2$, hence the greedy heuristic chooses $c_{i+1} = c_{i+1}^j$. Hence, sampling a point from K in the easy case can happen only constantly many times, in expectation, before K becomes covered after which we stop counting the hits to K.

But what do we do in the hard case? There, we at least know that with constant probability the cost drops by $\varphi(K, C_i)/2$, concretely we know that $\varphi(K, C_i) - E[\varphi(K, C_{i+1})] \geq \varphi(K, C_i)/5$. Recall that we are counting hits to K and each candidate center hits it with probability $\varphi(K, C_i)/\varphi(X, C_i)$.

Our situation is very similar to the following deterministic process where we start with a number X_0 (corresponding to $\varphi(X, C_1)$) and an empty counter $H_0 = 0$ (corresponding to counting hits). In each step we then choose some number $0 < K_i \leq X_i$ (corresponding to the cost of the cluster $\varphi(K, C_i)$) and define $X_{i+1} \leftarrow X_i - K_i$, while increasing the counter $C_{i+1} \leftarrow C_i + \ell \cdot K_i/X_i$. In this idealized process, it holds that $C_i = O(\ell \cdot \log X_0/X_i)$. Intuitively, this is because the case when $K_i = \Theta(X_i)$ in every step is the hardest one.

We can apply similar reasoning to our randomized process to get the expected bound $O\left(\ell \log \frac{OPT(1)}{OPT(k)}\right)$ on the number of hits. Here, we additionally use that 1) the starting cost of our solution $\varphi(X, C_1)$ is expected to be of order $OPT(1)$ by Lemma 2.1 and 2) the final cost $\varphi(X, C_k)$ has to be at least $OPT(k)$. $\qquad\square$