Asymmetric k-Center is $\log^* n$-Hard to Approximate

Julia Chuzhoy* Sudipto Guha† Eran Halperin‡ Sanjeev Khanna§

Guy Kortsarz¶ Robert Krauthgamer‖ Joseph (Seffi) Naor**

June 3, 2004

Abstract

In the Asymmetric k-Center problem, the input is an integer k and a complete digraph over n points together with a distance function obeying the directed triangle inequality. The goal is to choose a set of k points to serve as centers and to assign all the points to the centers, so that the maximum distance of any point to its center is as small as possible.

We show that the Asymmetric k-Center problem is hard to approximate up to a factor of $\log^* n - O(1)$ unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\log \log n})$. Since an $O(\log^* n)$-approximation algorithm is known for this problem, this resolves the asymptotic approximability of this problem. This is the first natural problem whose approximability threshold does not polynomially relate to the known approximation classes. We also resolve the approximability threshold of the metric (symmetric) k-Center problem with costs.

*Computer Science Dept., Technion, Haifa 32000, Israel. Email: cjulia@cs.technion.ac.il

†Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19103. Email:sudipto@cis.upenn.edu

‡International Computer Science Institute, Berkeley, CA 94704, USA and Computer Science Division, University of California, Berkeley, CA 94720, USA. Supported in part by NSF grants CCR-9820951 and CCR-0121555 and DARPA cooperative agreement F30602-00-2-0601. Email: eran@cs.berkeley.edu

§Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19103. Supported in part by an Alfred P. Sloan Research Fellowship and by an NSF Career Award CCR-0093117. Email:sanjeev@cis.upenn.edu

¶Department of Computer Sciences, Rutgers University, Camden, NJ 08102, USA. Email: guyk@camden.rutgers.edu

‖International Computer Science Institute, Berkeley, CA 94704, USA and Computer Science Division, University of California, Berkeley, CA 94720, USA. Supported in part by NSF grants CCR-9820951 and CCR-0121555 and DARPA cooperative agreement F30602-00-2-0601. Present address: IBM Almaden Research Center, 650 Harry Road, San Jose CA 95120, USA. Email: robi@almaden.ibm.com

**Computer Science Dept., Technion, Haifa 32000, Israel. Research supported in part by the United States-Israel Binational Science Foundation Grant No. 2002-276 and by EU contract IST-1999-14084 (APPOL II). Email: naor@cs.technion.ac.il

1 Introduction

The input to the Asymmetric k-Center problem consists of a complete digraph G with vertex set V, a non-negative weight (or distance) function $c_{uv} \geq 0$ for every $u, v \in V$, and an integer k. The weight function c satisies the directed triangle inequality, that is, $c_{uv} + c_{vw} \geq c_{uw}$ for all $u, v, w \in V$. Note that c_{uv} might differ from c_{vu}. The goal is to find a set S of k vertices, called *centers*, and to assign each vertex of V to a center, such that the maximal distance of a vertex from its center is minimized. More formally, we want to find a subset $S \subseteq V$ of size k, that minimizes

$$\max_{v \in V} \min_{u \in S} c_{uv}. \tag{1}$$

The quantity in (1) is called the *covering radius* of the centers S.

The problem is well-known to be NP-hard [11] and therefore, it is natural to seek approximation algorithms with small approximation ratio for the problem. If the function c is assumed to be symmetric as well, i.e. $c_{uv} = c_{vu}$ for all $u, v \in V$, the above problem is known as the (metric) k-Center problem. This is one of the early problems for which approximation algorithms were designed, and an optimal approximation ratio of 2 is known from the results of [6, 16, 13, 18, 20]. Subsequent to the solution of this problem a significant number of other problems in location theory were solved (see [23]); however, the approximability of the asymmetric case remained open[1], and was evoked by Shmoys [22].

For any positive integer n, define the iterated log function $\log^{(i)} n$ as follows: $\log^{(1)} n = \log n$ and $\log^{(i+1)} n = \log(\log^{(i)} n)$. (All logs are to the base 2.) The function $\log^* n$ is defined to be the least integer i for which $\log^{(i)} n \leq 1$. In a significant step, Panigrahy and Vishwanathan [19] designed an elegant $O(\log^* n)$ approximation algorithm for the Asymmetric k-Center problem, which was subsequently improved by Archer [3] to $O(\log^* k)$. Interestingly, [19] showed that given an Asymmetric k-Center instance, it is possible to compute in polynomial time a set of at most $2k$ centers whose value (covering radius) is within a factor of $\log^* \left(\frac{n}{k}\right)$ of the optimal solution with k centers. This approximation ratio tantalized researchers, partly because $\log^* n$ is an exotic function (in the area of approximation algorithms) and partly because it is so close to being a constant; nevertheless, no improved approximation algorithm was found.

We show that the approximation algorithms of [19, 3] are asymptotically best possible, unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\log \log n})$. This is a lower bound for a natural problem that does not conform to any of the known classes of approximation (see [1]). Recently, a sequence of papers [14, 15], has shown for the first time a natural problem(Group-Steiner-Tree) which is hard to approximate up to a poly-logarithmic factor. However, a hardness of $\log^* n$ is not even polynomially related to any of the known approximation classes.

1.1 Results

Our main result is a $\log^* n - O(1)$ hardness of approximation for the Asymmetric k-Center problem. More precisely, we show that:

[1]The problem is inapproximable if the triangle inequality does not hold.

- There is a constant $\alpha > 0$ such that Asymmetric k-Center cannot be approximated within a factor of $\log^* n - \alpha$, unless $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\log\log n})$.
- The above result holds also for bicriteria algorithms, that are allowed to use $O(k)$ additional centers while their solution is compared against an optimum that uses only k centers.

Previously, the only hardness result known was 2, which follows immediately from the symmetric case.

Finally, we show that the (metric) k-Center problem with (non-uniform) vertex costs is hard to approximate within a factor better than 3. This matches the 3–approximation of Hochbaum and Shmoys [16], and separates the problem from its uniform cost counterpart (which has 2–approximation).

1.2 Techniques

Our results build on a sequence of recent papers leading to a hardness of $(d - 1 - \epsilon)$ on the approximation factor for d-Hypergraph Cover [17, 10, 9, 7, 8] (the vertex cover problem on hypergraphs, where each hyperedge contains exactly d vertices). In order to optimize our leading constant we use one of the results of Dinur, Guruswami and Khot [7], which they call "the simple construction". This result can be viewed as a construction of an instance of Set-Cover from an instance of a Gap-3SAT(5) problem – the hypergraph vertices correspond to sets while the hypergraph edges correspond to elements. As shown by Arora *et al.* [2], there exists some $0 < \epsilon < 1$, such that it is NP-hard to decide whether an instance of Gap-3SAT(5) is a yes-instance (the input formula is satisfiable) or a no-instance (at most a fraction $(1 - \epsilon)$ of the clauses are simultaneously satisfiable). It can be shown that the construction of [7] achieves a strong bicriteria gap: If the input 3SAT(5) formula is a yes-instance then an $O(1/d)$-fraction of the sets are sufficient to cover all the elements. If the formula is a no-instance then any collection of $(1 - 2/d)$-fraction of the sets covers at most a $(1 - f(d))$-fraction of the elements with $f(d) = 1/2^{\mathrm{poly}(d)}$. Suppose we were to "compose" it with another Set-Cover instance, in the sense that the elements of the first instance are actually the sets of the second instance. Then any $(1 - 2/d)$-fraction of the sets in the first instance covers at most $(1 - f(d))$-fraction of the sets of the second instance. If the second Set-Cover instance is constructed using $d' = 2/f(d)$, then the already covered sets of the second instance are not sufficient to cover all the elements of the second instance. In other words, no $(1 - 2/d)$-fraction of the sets in the first instance can cover "within distance 2" all the elements of the second instance. This process can be continued further, with the limitation being the rapid growth in the construction size since the value of d in successive instances must grow as $2^{\mathrm{poly}(d)}$.

More specifically, our reduction works as follows. Given an instance ψ of size n for Gap-3SAT(5), we build a directed graph with $N = O(n^{\log\log n})$ vertices. The graph vertices are partitioned into $h + 2$ layers, where $h = \log^* n - \Theta(1) = \log^* N - \Theta(1)$. For each pair of consecutive layers, i and $i + 1$, there are directed edges from some layer i vertices to some layer $(i+1)$ vertices. This graph is transformed into an instance of Asymmetric k-Center as follows. The set of vertices remains the same, the distance c_{uv} is the length of the shortest (directed) path from u to v, and k is set a certain value.

Layer 0 of the vertex set consists of only one vertex which is connected to every vertex in layer 1. For any two other consecutive layers, i and $i+1$, we build a SET-COVER instance, where layer i vertices serve as sets, and layer $(i+1)$ vertices serve as elements. There is a directed edge from layer i vertex v to layer $i+1$ vertex u if and only if the element corresponding to u belongs to the set corresponding to v.

If the formula ψ is a yes-instance, all the vertices can be covered by k centers with radius 1, essentially by taking taking the solutions to all the SET-COVER instances, using in total only $k-1$ sets (vertices), and adding the vertex at level 0.

If ψ is a no-instance, we prove that it is impossible to cover all the vertices by k centers with radius h. To do this, it is enough to show that it is impossible to choose $k-1$ vertices in layer 1 that cover (with radius h) all the vertices in layer $h+1$. Indeed, we can assume that every solution uses only vertices in layers 0 and 1, since any solution must contain the layer 0 vertex (because it is impossible to cover this vertex otherwise), and this vertex covers (with radius h) all the vertices except for layer $h+1$. As we are allowed to use radius h, there is no point in taking any vertex v of some layer $i > 1$ to the solution – any predecessor of v in layer 1 can cover all the vertices v can cover.

Organization

The rest of the paper is organized as follows. Section 2 presents the bicriteria hardness for SET-COVER that we require. The reduction to ASYMMETRIC k-CENTER is given in Section 3. The hardness proof also provides an explicit construction of an integrality gap of $\log^* n - O(1)$ for the linear program used by Archer [3]. In Section 4 we show tight lower bounds for the (metric) k-CENTER problem with (non-uniform) vertex costs.

2 A Bicriteria Hardness Result for Hypergraph Cover

In this section we set up the stepping stone for the hardness of ASYMMETRIC k-CENTER problem. We will use the d-HYPERGRAPH COVER problem which is defined as follows. Given a set of M vertices and a collection of N hyperedges (i.e., subsets of vertices) of cardinality d, the goal is to find a minimum size set of vertices S such that every hyperedge contains at least one vertex from S. This problem can also be viewed as a SET-COVER instance where the vertices of the hypergraph correspond to sets and the hyperedges correspond to elements, so each element belongs to exactly d sets. (In SET-COVER, the input is a collection of sets of so-called elements, and the goal is to find a minimum number of sets whose union equals the union of all the input sets.)

The reduction is performed from the GAP-3SAT(5) problem, which is defined as follows. The input is a CNF formula ψ on n variables and $\frac{5n}{3}$ clauses. Each clause contains exactly 3 literals and each variable appears in 5 different clauses. Formula ψ is called a yes-instance if it is satisfiable. It is called a no-instance (with respect to some ϵ) if at most a fraction $(1-\epsilon)$ of clauses are simultaneously satisfiable. As shown by Arora *et al.* [2], there exists $0 < \epsilon < 1$ such that it is NP-hard to distinguish between the yes and the no instances of the problem.

The goal of this section is to prove the following theorem:

Theorem 2.1. *Given a* Gap-3SAT(5) *formula ψ and integer d, we can construct a d-*Hypergraph Cover *instance with the following properties:*

- *If ψ is a yes-instance, then all the hyperedges in the hypergraph can be covered using a fraction $\frac{3}{d}$ of the vertices.*
- *If ψ is a no-instance, then no subset containing at most a $(1-\frac{1}{d})$-fraction of the vertices covers all the hyperedges.*
- *The hypergraph size is $n^{O(\log d)}2^{d^\beta}$ for some sufficiently large constant $\beta \geq 1$, and it can be constructed in time polynomial in it its size. Moreover, if M denotes the number of vertices and N is the number of hyperedges, then $N \leq 2^{d^\beta} M$.*

We note that the above theorem follows directly from [7, 8]. The reduction presented below is identical to the one called "simple construction" in [7]. However, we find it more convenient to change the parameter p of the construction (which is explained below) to $(1 - \frac{3}{d})$, as well as to use [8] to bound the size of s-wise t-intersecting families. We provide the construction for the sake of completeness and also because we use some of its properties which are not proven explicitly in [7, 8].

2.1 s-wise t-intersecting families

Suppose we are given a ground set R. A family $\mathcal{F}$ of subsets of R is called s-wise t-intersecting if for every collection of s sets $F_1, F_2, ..., F_s \in \mathcal{F}$, we have $|F_1 \cap F_2 ... \cap F_s| \geq t$. Following [7, 8], define the weight of a set $F \subseteq R$ to be $p^{|F|}(1-p)^{|R\setminus F|}$, i.e., the probability of obtaining F when each element of R is chosen independently at random with probability p. The weight of a collection $\mathcal{F}$ of sets is defined to be the sum of the weights of the sets in the collection.

Lemma 2.1 (Lemma 2.5 of [8]). *Let s, t be some integers, and let $p < 1 - \frac{1}{s}$. Then, the weight of any s-wise t-intersecting family is at most*

$$\frac{e^{-2t(1-\frac{1}{s}-p)^2}}{1-e^{-2s(1-\frac{1}{s}-p)^2}}$$

Setting $s = \frac{d}{2}, p = 1 - \frac{3}{d}$, the bound simplifies to $\frac{e^{-2t/d^2}}{1-e^{-1/d}}$. Using $1 - e^{-x} \geq \frac{x}{2}$ for $0 \leq x \leq \frac{1}{2}$, the bound becomes $2de^{-2t/d^2}$.

Corollary 2.2. *Let d be an even integer, $p = 1 - \frac{3}{d}$ and $t = 4d^2 \ln d$. Then, the weight of any $\frac{d}{2}$-wise t-intersecting family is at most $\frac{1}{2d}$.*

2.2 The d-Hypergraph Cover Hardness

Our starting point is the Raz Verifier for Gap-3SAT(5) with ℓ repetitions, which is defined as follows. Given an instance ψ of size n for Gap-3SAT(5), the verifier chooses independently at random l clauses $C_1, \ldots, C_\ell$ from ψ. In each clause C_i, $1 \leq i \leq \ell$, one variable α_i (called a distinguished variable) is chosen. Prover 1 receives the collection of clauses

$C_1, \ldots, C_\ell$ and is expected to answer with an assignment to all the variables appearing in the clauses, and prover 2 receives the collection of distinguished variables $\alpha_1, \ldots, \alpha_\ell$ and is expected to answer with an assignment to all the distinguished variables. The verifier then checks that the assignment of prover 1 satisfies all the clauses and that the answers of the two provers are consistent (imply same values for the distinguished variables).

Let X and Y denote the collections of all the possible queries of prover 1 and 2 respectively. Given query $x \in X$, let R_x be the set of all the possible answers of prover 1 that satisfy all the clauses in x. Clearly, $|X| = n^{O(l)}$ and for all $x \in X$, $|R_x| = 7^l$. Similarly, for each $y \in Y$, R_y denotes the set of all the possible answers of prover 1 to query y. Each random string r defines a constraint φ which depends on the queries $x \in X$, $y \in Y$ corresponding to r. Note that for every $a_x \in R_x$ assigned to x there is exactly one value $a_y \in R_y$ that satisfies the constraint φ. For convenience, the constraint φ is viewed as a function $\varphi_{x \to y} : R_x \to R_y$. The set of constraints is denoted by Φ. Note that every $x \in X$ appears in exactly 3^ℓ constraints and every $y \in Y$ appears in 5^ℓ constraints.

Theorem 2.2 ([4, 2, 21]). *There exists a constant $\gamma > 0$ such that for all n and ℓ, the above set of constraints Φ satisfies:*

- *If ψ is a yes-instance, then there is an assignment that satisfies all the constraints.*
- *If ψ is a no-instance, then no assignment satisfies more than a $2^{-\gamma\ell}$ fraction of the constraints.*

Given a Gap-3SAT(5) instance ψ and an even d, we build a d-hypergraph $H = (V, E)$. The vertex set is $V = \{\langle x, F\rangle \mid x \in X, F \subseteq R_x\}$. The set of hyperedges is defined as follows. Suppose $x, x' \in X$, such that for some $y \in Y$, $\varphi_{x \to y}, \varphi_{x' \to y} \in \Phi$. Let $a \in R_x$, $a' \in R_{x'}$ be some assignments to x, x'. We say that these assignments are consistent if they imply the same assignment to every y' such that $\varphi_{x \to y'}, \varphi_{x' \to y'} \in \Phi$, that is, $\varphi_{x \to y'}(a) = \varphi_{x' \to y'}(a')$.

Now, for any pair $x, x' \in X$, such that for some $y \in Y$, $\varphi_{x \to y}, \varphi_{x' \to y} \in \Phi$, consider any d vertices of the form $\langle x, A_1\rangle, \langle x, A_2\rangle, \ldots, \langle x, A_{\frac{d}{2}}\rangle$ and $\langle x', B_1\rangle, \langle x', B_2\rangle, \ldots, \langle x', B_{\frac{d}{2}}\rangle$. Then, there is a hyperedge between these vertices if and only if there is no pair of consistent assignments $a \in \bigcap_{i=1}^{\frac{d}{2}} A_i$, $a' \in \bigcap_{i=1}^{\frac{d}{2}} B_i$. The following observation about the construction is useful:

Proposition 2.3. *Consider a collection $\langle x, A_1\rangle, \ldots, \langle x, A_{\frac{d}{2}}\rangle$, $\langle x', B_1\rangle, \ldots, \langle x', B_{\frac{d}{2}}\rangle$ of d vertices (d is even). Suppose that for some y the constraints $\varphi_{x \to y}, \varphi_{x' \to y}$ exist and there is no hyperedge containing the d vertices $\langle x, A_1\rangle, \ldots, \langle x, A_{\frac{d}{2}}\rangle$ and $\langle x', B_1\rangle, \ldots, \langle x', B_{\frac{d}{2}}\rangle$. Then, there must be an $a_x \in \bigcap_{j=1}^{\frac{d}{2}} A_j$ and an $a_{x'} \in \bigcap_{j=1}^{\frac{d}{2}} B_j$ such that assigning a_x to x and $a_{x'}$ to x' is consistent with some assignment b to y.*

For every subset $A \subseteq R_X$, define its weight to be the probability of choosing it if each element of R_X is chosen independently with probability $p = 1 - \frac{3}{d}$. The weight of a vertex $\langle x, A\rangle$ is the weight of A, i.e., $p^{|A|}(1-p)^{|R_X \setminus A|}$. The next lemma follows by choosing all vertices $\langle x, F\rangle$ for which F does not include the correct assignment to x.

Lemma 2.4 (Lemma 3.5 in [7]). *If Φ is satisfiable then there exists a hypergraph cover of weight at most $(1-p)|X| = (3/d)|X|$.*

The next lemma follows from the contrapositive of Corollary 2.2, that is, if a collection of sets $\mathcal{A}$ has large weight, then there must be $s = \frac{d}{2}$ sets in the collection whose intersection is at most $t = 4d^2 \ln n$.

Lemma 2.5 (Implicit in proof of Lemma 3.6 [7]). *Suppose we are given a collection $\mathcal{A}$ of subsets of R_X. If the set of vertices $\{\langle x, F\rangle | F \in \mathcal{A}\}$ has weight greater than $\frac{1}{2d}$, then there are $\frac{d}{2}$ sets $A_x(1), \ldots, A_x(\frac{d}{2})$ in the collection $\mathcal{A}$ such that $\left|\bigcap_{j=1}^{\frac{d}{2}} A_x(j)\right| \leq t = 4d^2 \ln d$.*

Lemma 2.6 ([7], Lemma 3.6). *If there exists a hypergraph cover of weight less than $(1 - \frac{1}{d})|X|$, then we can satisfy $\frac{1}{32d^5 \ln^2 d}$ of the constraints Φ.*

Proof. The proof follows the proofs of Proposition 3.4 and Lemma 3.6 in [7]. We present the proof here for the sake of completeness.

Fix a cover of the hypergraph. For each variable x, let $I(x)$ be the set of vertices $\langle x, A\rangle$, $A \subseteq R_x$, which are not in the cover. Define X' to be the set of variables $x \in X$ for which the weight of $I(x)$ is greater than $\frac{1}{2d}$. It follows from a simple averaging argument that at least $\frac{1}{2d}$ fraction of the variables in X belong to X'. From now on, we focus only on the variables in X'. Since each variable in X participates in the same number of the original $x \to y$ constraints, the variables in X' participate in at least a fraction $\frac{1}{2d}$ of the constraints in Φ.

For each $x \in X'$ define $\mathcal{A}_x = \{F | \langle x, F\rangle \in I(x)\}$. By Lemma 2.5, there exist sets $A_x(1), \ldots, A_x(\frac{d}{2})$ in $\mathcal{A}_x$ such that

$$\left|\bigcap_{i=1}^{\frac{d}{2}} A_x(i)\right| \leq t = 4d^2 \ln d$$

Define $T(x) = \bigcap_{i=1}^{\frac{d}{2}} A_x(i)$. We show an assignment to $X \cup Y$ that satisfies a large fraction of constraints. For $x \in X'$, pick any t assignments in $T(x)$ randomly as an assignment for x.

For a variable $y \in Y$, pick an arbitrary $x_y \in X'$ such that the constraint $\varphi_{x_y \to y}$ exists. Choose a random element $a \in T(x_y)$ and give y the assignment $\varphi_{x_y \to y}(a)$.

Now, let us evaluate the fraction of constraints $\{\varphi_{x \to y} | x \in X'\}$ which are satisfied. There are two cases to consider. If $x = x_y$, then the probability we satisfy $\varphi_{x \to y}$ is $\frac{1}{t}$. Otherwise, if $x \neq x_y$, we claim that there must be an assignment $a \in T(x)$ and $a' \in T(x_y)$, such that assigning a to x and a' to x_y implies the same assignment to y. This is true since there is no hyperedge spanning the d vertices $\langle x, A_1(x)\rangle, \ldots, \langle x, A_{\frac{d}{2}}(x)\rangle$, and $\langle x_y, A_1(x_y)\rangle$, $\ldots$, $\langle x_y, A_{\frac{d}{2}}(x_y)\rangle$ (otherwise, it would contradict that we have a cover), and thus we can invoke Proposition 2.3. Now, the probability that y was assigned a value consistent with the assignment of a' to x_y is $\frac{1}{t}$, and furthermore the probability that x was assigned the value a is $\frac{1}{t}$. Therefore, with probability at least $\frac{1}{t^2}$ the constraint $\varphi_{x \to y}$ is satisfied.

Since the fraction of constraints involving variables in X' is at least $\frac{1}{2d}$, the expected fraction of satisfied constraints in Φ is at least $\frac{1}{32d^5 \ln^2 d}$. Thus, the lemma follows. □

Setting $\ell = \Theta(\log d)$, so that $\frac{1}{32d^5 \ln^2 d} > 2^{-\gamma\ell}$ holds, we ensure that for a no-instance, no cover of weight less than $(1 - \frac{1}{d})|X|$ exists.

The above constructs a weighted instance of a hypergraph cover. The number of vertices in the construction is $M = |X| \cdot 2^{7^\ell}$ and the number of edges is $N \leq |X| \cdot 15^\ell \cdot 2^{7^\ell d}$ (since for each $x \in X$, there are at most 15^ℓ queries $x' \in X$ such that $\varphi_{x \to y}, \varphi_{x' \to y} \in \Phi$ for some $y \in Y$). The instance can be converted into an unweighted instance by replicating vertices appropriately along the lines of [10, 7, 8]. This will increase the construction size by a factor of $2^{\text{poly}(d)}$. Therefore, for some sufficiently large positive integer β, the size of the construction is bounded by $n^{O(\log d)} \cdot 2^{d^\beta}$, and $N \leq 2^{d^\beta} M$.

This completes the proof of Theorem 2.1.

Corollary 2.7. *In the above hypergraph, in the no-instance case, no subset containing at most a* $(1 - \frac{2}{d})$*-fraction of the vertices covers more than a* $1 - \frac{1}{d2^{d^\beta}}$ *fraction of the hyperedges.*

Proof. Assume by contradiction that we can choose a $1 - \frac{2}{d}$ fraction of the vertices that covers a fraction $1 - \frac{1}{d2^{d^\beta}}$ of the hyperedges. We can then cover the remaining hyperedges by using one additional vertex for each edge. But, since $N/d2^{d^\beta} \leq M/d$, we would be using less than $(1 - \frac{2}{d})M + \frac{M}{d} = (1 - \frac{1}{d})M$ vertices to cover all the hyperedges, which contradicts Theorem 2.1. □

In what follows, we refer to the Set-Cover instances used in the above corollary as the *basic* Set-Cover *instances with parameter* d.

3 Hardness of Asymmetric k-Center

We now use the machinery of Section 2 to present our hardness result for Asymmetric k-Center.

3.1 The reduction

We use the basic Set-Cover instances to build a directed graph with $h+2$ layers of vertices. For each pair of consecutive layers, i and $i+1$, there are directed edges from layer i vertices to layer $i+1$ vertices corresponding to an encoding of basic Set-Cover instances with suitably chosen parameters. This graph is transformed into an instance of Asymmetric k-Center as follows. The set of vertices remains the same and the distance $c(v, u)$ is the length of the shortest path from v to u.

Layer 0 of vertices consists of only one vertex, which is connected to each vertex in layer 1. For each pair of consecutive layers, i and $i+1$, $1 \leq i \leq h$, we use multiple disjoint copies of the basic Set-Cover instance, denoted by SC_i, as constructed in Section 2, with a parameter d_i that will be chosen soon. In this Set-Cover instance, the sets are represented by the vertices of layer i and the elements are represented by vertices of layer $i+1$. There is a directed edge from vertex v in layer i to vertex u in layer $i+1$ if and only if the element corresponding to u belongs to the set corresponding to v.

We define the parameters d_i inductively as follows. d_1 is chosen to be any positive integer greater than 6, and $d_{i+1} = 2^{d_i^{2\beta}}$ (where β is the constant from Section 2). The number of layers h is the maximum integer for which $d_h \leq \log^{(3)} n$ holds.

Let M_i and N_i denote the number of sets and elements in the basic Set-Cover instance with parameter d_i. Layer i Set-Cover instance, SC_i, consists of c_i disjoint copies of the basic Set-Cover instance with parameter d_i. Since the vertices of layer i are both the sets of SC_i and the elements of SC_{i-1}, we need to ensure that $c_i M_i = c_{i-1} N_{i-1}$. To this end, we set $c_i = \prod_{j=1}^{i-1} N_j \cdot \prod_{j=i+1}^{h} M_j$. Let the number of vertices in layer i be denoted by V_i. The number of vertices in layer 1 is therefore $V_1 = c_1 M_1 = \prod_{j=1}^{h} M_j$. Finally, set $k = 4V_1/d_1 + 1$.

Proposition 3.1. *For all $i \geq 1$, $\beta \geq 3$, and $d_1 \geq 3$,*

$$\log^{(i)} d_i \leq 3\beta \log d_1.$$

Proof. For every $i \geq 2$, it suffices to prove that $\log^{(j)} d_i \leq d_{i-j}^{3\beta}$ for all $1 \leq j \leq i-1$. We prove this by induction on j. The case $j = 1$ is immediate. For the inductive step, observe that $\log^{(j+1)} d_i \leq \log(d_{i-j}^{3\beta}) = 3\beta d_{i-j-1}^{2\beta}$, where the inequality is due to the induction hypothesis for j and the equality is by definition of d_{i-j}. Since $d_1 \geq 3$ and $\beta \geq 3$, we have $3\beta \leq d_1^{\beta} \leq d_{i-j-1}^{\beta}$, which yields the desired $\log^{(j+1)} d_i \leq d_{i-j-1}^{3\beta}$. □

Thus, for some constant γ we have $\log^* d_h \leq h + \gamma$. So whenever $h \leq \log^* n - 3 - \gamma$, we have $\log^* d_h \leq \log^* n - 3$ and thus $d_h \leq \log^{(3)} n$ holds. Therefore, choosing h as the maximum integer for which $d_h \leq \log^{(3)} n$ results in $h \geq \log^* n - O(1)$. It is also easy to see that $h \leq \log^* n$.

The size of the construction

The total number of vertices in this instance is

$$\begin{aligned} |V| = \sum_{i=1}^{h+1} V_i &\leq h \left(\prod_{i=1}^{h} N_i \cdot \prod_{i=1}^{h} M_i \right) \\ &\leq h \prod_{i=1}^{h} \left(n^{O(\log d_i)} 2^{d_i^{\beta}} \right)^2 \\ &\leq h \cdot n^{O(h \log d_h)} \cdot 2^{2h d_h^{\beta}} \leq n^{\log \log n}. \end{aligned}$$

Notice that $\log^* n = \log^* |V| - \Theta(1)$, and so $h = \log^* |V| - \Theta(1)$ as well.

3.2 Analysis of the reduction

We now show that our reduction to Asymmetric k-Center creates a gap between a yes-instance and a no-instance.

Lemma 3.2 (Yes-Instance). *Suppose ψ is a yes-instance. Then, $k = 4V_1/d_1 + 1$ centers can cover all the vertices with radius* 1.

Proof. Consider the following centers. At layer 0 take the single vertex, and at every layer $1 \leq i \leq h$ take $k_i = c_i \frac{3M_i}{d_i} = \frac{3V_i}{d_i}$ vertices according to the solution of SC_i (which is c_i disjoint basic Set-Cover instances). Clearly, these centers cover every vertex in V within radius 1.

To bound the number of centers, we first show that the sequence k_i decreases geometrically, namely, $k_i \leq \frac{k_{i-1}}{d_1}$. Indeed, for all $i \geq 2$,

$$\begin{aligned}
\frac{k_i}{k_{i-1}} &= \frac{3V_i}{d_i} \cdot \frac{d_{i-1}}{3V_{i-1}} \\
&\leq \frac{c_{i-1}N_{i-1}}{c_{i-1}M_{i-1}} \cdot \frac{d_{i-1}}{d_i} \quad \text{(since } V_{i-1} = c_{i-1}M_{i-1} \text{ and } V_i = c_{i-1}N_{i-1}\text{)} \\
&\leq 2^{d_{i-1}^{\beta}} \cdot \frac{d_{i-1}}{d_i} \quad \text{(since } N_{i-1} \leq 2^{d_{i-1}^{\beta}} M_{i-1}\text{)} \\
&< \frac{1}{d_1} \quad \text{(since } d_i = 2^{d_{i-1}^{2\beta}}\text{)}.
\end{aligned}$$

Therefore, the total number of vertices we use in the solution is $k = 1 + \sum_i k_i < 1 + k_1(1 + \frac{1}{d_1 - 1}) \leq 1 + \frac{4V_1}{d_1}$. (The last inequality assumes $d_1 \geq 4$.) ☐

Lemma 3.3 (No-Instance). *If the formula ψ is a no-instance, then it is impossible to cover all the vertices with radius h, using $k = 4V_1/d_1 + 1$ centers, for $d_1 \geq 7$.*

To prove this lemma, it suffices to show that no $k-1$ vertices in layer 1 can cover (with radius h) all the vertices in layer $h+1$. Indeed, any solution must contain the vertex in layer 0 (as this is the only way to cover it), and this vertex covers within radius of h all the vertices except for those in layer $h+1$. In order to cover the layer $h+1$ vertices (with radius h), there is no point selecting centers in any layer other than 1, since for any center v in a layer $i > 1$, we can cover the same vertices by choosing a predecessor of v in layer 1. (It is easy to see there always exists one.) Therefore, the proof of Lemma 3.3 follows immediately from the next claim.

Claim 3.4. *Let S be a set of $k-1$ centers in layer 1. Then, in every layer $i \geq 1$, the fraction of vertices unreachable from S is at least $\delta_i = 3/d_i$, assuming $d_1 \geq 7$.*

Proof. We proceed by induction on i. For $i = 1$ this is clear since the fraction of vertices in layer 1 that are not in the solution is $1 - \frac{k-1}{V_1} = 1 - \frac{4}{d_1} \geq \frac{3}{d_1}$ for all $d_1 \geq 7$. Consider now $i \geq 1$, and assume the fraction of vertices in layer i that are reachable from S is at most $1 - \delta_i$.

Consider the Set-Cover instance SC_i. The fraction of vertices in V_i (the sets for SC_i) that are reachable from S is at most $1-\delta_i$. The fraction of basic Set-Cover instances in SC_i in which these sets constitute more than a $1 - \frac{2}{d_i}$ fraction is thus at most $(1 - \frac{3}{d_i})/(1 - \frac{2}{d_i}) = 1 - \frac{1}{d_i - 2}$. The remaining basic Set-Cover instances comprise at least a fraction of $\frac{1}{d_i-2}$ of the c_i basic instances in SC_i. In each of these, at least a fraction of $1/d_i 2^{d_i^{\beta}}$ of the elements are not reachable from S by Corollary 2.7. Thus the total fraction of vertices of layer $i+1$ that are unreachable from S is at least

$$\frac{1}{d_i - 2} \cdot \frac{1}{d_i 2^{d_i^{\beta}}} \geq \frac{3}{d_{i+1}}.$$

☐

Our main result now follows from Lemmas 3.2 and 3.3 (in conjunction with Section 2).

Theorem 3.1. Asymmetric k-Center *cannot be approximated within ratio* $\log^* n - \alpha$ *for some constant* α, *unless* $\mathsf{NP} \subseteq \mathsf{DTIME}(n^{\log\log n})$.

We note that for any constant h, our construction implies that there is no h-approximation for Asymmetric k-Center, under the weaker assumption of $\mathsf{P} \neq \mathsf{NP}$. We also note that by choosing a suitably larger value of the constant d_1, we can obtain the hardness result of Them 3.1 even when the approximation algorithm is allowed to use $a \cdot k$ centers for any constant $a \geq 1$.

3.3 Integrality Gap

Our reduction also provides an explicit construction of an integrality gap of $\log^* n - \Omega(1)$ with respect to the linear program used by Archer [3].[2] Indeed, in a no-instance any integral solution, i.e., k centers, has value (covering radius) $\log^* n - \Omega(1)$. On the other hand, the reduction of [7] constructs a d-Hypergraph Cover instance, and thus every vertex in layer $i+1$ in our construction is adjacent to exactly d_i vertices in layer i. It follows that a fractional solution, where every vertex at layer i is taken to be a center to the extent of $\frac{1}{d_i}$, covers all the vertices of layer $i+1$ within distance 1. (See [3] for the precise linear program formulation.) Hence, all vertices in all the layers can be fractionally covered within a distance 1, and the total number of fractional centers is (similar to the yes-instance) only $1 + \sum_i \frac{V_i}{d_i} \leq \frac{k}{3}$.

This integrality gap instance construction does not actually require the reduction of [7]. We can simply replace every SC_i instance by a random d-Hypergraph Cover instance, i.e., let every vertex in layer $i+1$ have incoming edges from d_i (distinct) random vertices in layer i. It can be verified, using a union bound, that with high probability the resulting d-Hypergraph Cover instance satisfies the properties that we require from Section 2.

4 Implications for Symmetric Distance Functions

The same reduction (but with $h = 2$) shows another interesting hardness result for metric k-Center with costs (sometimes called weighted k-center). In this problem we are given a distance metric c over the vertices, a nonnegative cost function w for the vertices, and a cost bound k. (Note that being a metric, c is symmetric.) The goal is to choose a subset S of the vertices having total cost at most k so as to minimize

$$\max_{v \in V} \ \min_{u \in S} \ c_{uv}. \tag{2}$$

Here, too, the vertices of S are called *centers* and the quantity in (2) is called the *covering radius* of S.

Hochbaum and Shmoys [16] show a 3-approximation algorithm for this problem. In what follows we show that this bound is tight. In contrast, if all vertices have unit cost then the

[2]This is under a slightly nonstandard notion of integrality gap, because the linear program is actually not a relaxation of Asymmetric k-Center.

problem specializes to the familiar metric k-CENTER problem, which has a 2–approximation. If we were allowed to discard a small fraction of the vertices (in the metric k-center with costs), lower and upper bounds of 3 are known [5].

Theorem 4.1. *It is NP-hard to approximate the metric k-CENTER problem with costs to a factor less than 3.*

Proof. We construct the same layered instance as in ASYMMETRIC k-CENTER, but with $h = 2$. Since the number of layers is constant, the instance can be constructed in polynomial time. However, the edges in this case are undirected.

The vertices in the last layer ($h + 1 = 3$) have arbitrarily large weight (greater than k suffices) to rule out choosing them in any solution. The weight of any other vertex is 1.

If the formula ψ is a yes-instance, then by Lemma 3.2 we can cover all the vertices within radius 1 using at most $4V_1/d_1$ centers from layers $0, 1$ and 2.

If ψ is a no-instance, then for the purpose of covering layer 3 within radius 2, we can replace any center in layer 2 with a neighbor of it from layer 1. And by Lemma 3.3 we know that by allocating the entire budget to centers in layer 1, one cannot cover all the vertices in layer 3 within radius 2. Hence, no set of centers of total cost k can cover all of layer 3 with radius smaller than 3. □

Acknowledgements

We would like to thank Samir Khuller for bringing the metric k-CENTER with costs to our attention. We would like to thank Venkat Guruswami and Howard Karloff for their comments on previous drafts of the paper. We would also like to thank the anonymous reviewers whose valuable comments have helped improve our presentation.

References

[1] S. Arora and C. Lund. Hardness of approximation. In Dorit Hochbaum, editor, *Approximation Algorithms for NP Hard Problems*. PWS publishing Co, 1996.

[2] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *JACM*, 45(3):501–555, 1998.

[3] A. F. Archer. Two $O(\log^* k)$-approximation algorithms for the asymmetric k-center problem. *Proceedings of the 8th Conference on Integer Programming and Combinatorial Optimization*, pp. 1–14, 2001.

[4] S. Arora and S. Safra. Probabilistic checking of proofs: A new characterization of NP. *JACM*, 45(1):70–122, 1998.

[5] M. Charikar, S. Khuller, D. M. Mount, and G. Narasimhan. Algorithms for facility location problems with outliers. *Proceedings of the 12th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 642–651, 2001.

[6] M. E. Dyer and A. M. Frieze. A simple heuristic for the p-center problem. *Oper. Res. Lett.*, 3(6):285–288, 1985.

[7] I. Dinur, V. Guruswami, and S. Khot. Vertex cover on k-uniform hypergraphs is hard to approximate within factor $(k-3-\epsilon)$. *Electronic Colloquium on Computational Complexity (ECCC)*, (027), 2002.

[8] I. Dinur, V. Guruswami, S. Khot, and O. Regev. A new multilayered PCP and the hardness of hypergraph vertex cover. *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pp. 595–601, 2003.

[9] I. Dinur, O. Regev, and C. D. Smyth. The hardness of 3-uniform hypergraph coloring. *Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science*, 2002.

[10] I. Dinur and S. Safra. The importance of being biased. *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pp. 33–42, 2002.

[11] M. R. Garey and D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-completeness. W.H. Freeman and Company, 1979.

[12] O. Goldreich. Using the FGLSS-reduction to Prove Inapproximability Results for Minimum Vertex Cover in Hypergraphs. *Electronic Colloquium on Computational Complexity (ECCC)* 102, 2001.

[13] T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoret. Comput. Sci.*, 38(2-3):293–306, 1985.

[14] E. Halperin, G. Kortsarz, R. Krauthgamer, A. Srinivasan, and N. Wang. Integrality ratio for group steiner trees and directed steiner trees. *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 275-284, 2003.

[15] E. Halperin and R. Krauthgamer. Polylogarithmic inapproximability. *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pp. 585–594, 2003.

[16] D. S. Hochbaum and D. B. Shmoys. A unified approach to approximation algorithms for bottleneck problems. *JACM*, 33(3):533–550, 1986.

[17] J. Holmerin. Vertex cover on 4-regular graphs is hard to approximate within $2-\epsilon$. *Proceedings of the 34th ACM Symposium on Theory of Computing*, pp. 544–552, 2002.

[18] W. L. Hsu and G. L. Nemhauser. Easy and hard bottleneck location problems. *Discrete Applied Math.*, 1(3):209–215, 1979.

[19] R. Panigrahy and S. Vishwanathan. An $O(\log^* n)$ approximation algorithm for the asymmetric p-center problem. *J. of Algorithms*, 27(2):259–268, 1998.

[20] J. Plesník. On the computational complexity of centers locating in a graph. *Aplikace Matematiky*, 25(6):445–452, 1980.

[21] R. Raz. A parallel repetition theorem. *SIAM J. of Computing*, 27(3):763–803, 1998.

[22] D.B. Shmoys. Computing near-optimal solutions to combinatorial optimization problems. In *Combinatorial Optimization*, (W. Cook, L. Lovasz, and P.D. Seymour, eds.), pp.355–397, AMS, 1995.

[23] V. V. Vazirani. *Approximation Algorithms*. Springer Verlag, 2001.