Coresets for Clustering in Geometric Intersection Graphs

Sayan Bandyapadhyay
Portland State University, OR, USA

Fedor V. Fomin
University of Bergen, Norway

Tanmay Inamdar
University of Bergen, Norway

Abstract

Designing coresets – small-space sketches of the data preserving cost of the solutions within $(1 \pm \epsilon)$-approximate factor – is an important research direction in the study of center-based k-clustering problems, such as k-means or k-median. Feldman and Langberg [STOC'11] have shown that for k-clustering of n points in general metrics, it is possible to obtain coresets whose size depends logarithmically in n. Moreover, such a dependency in n is inevitable in general metrics. A significant amount of recent work in the area is devoted to obtaining coresests whose sizes are independent of n for special metrics, like d-dimensional Euclidean space [Huang, Vishnoi, STOC'20], doubling metrics [Huang, Jiang, Li, Wu, FOCS'18], metrics of graphs of bounded treewidth [Baker, Braverman, Huang, Jiang, Krauthgamer, Wu, ICML'20], or graphs excluding a fixed minor [Braverman, Jiang, Krauthgamer, Wu, SODA'21].

In this paper, we provide the first constructions of coresets whose size does not depend on n for k-clustering in the metrics induced by *geometric intersection graphs*. For example, we obtain $\frac{k \log^2 k}{\epsilon^{\mathcal{O}(1)}}$ size coresets for k-clustering in Euclidean-weighted unit-disk graphs (UDGs) and unit-square graphs (USGs). These constructions follow from a general theorem that identifies two canonical properties of a graph metric sufficient for obtaining coresets whose size is independent of n. The proof of our theorem builds on the recent work of Cohen-Addad, Saulpic, and Schwiegelshohn [STOC '21], which ensures small-sized coresets conditioned on the existence of an interesting set of centers, called *centroid set*. The main technical contribution of our work is the proof of the existence of such a small-sized centroid set for graphs that satisfy the two canonical properties. Loosely speaking, the metrics of geometric intersection graphs are "similar" to the Euclidean metrics for points that are close, and to the shortest path metrics of planar graphs for points that are far apart. The main technical challenge in constructing centroid sets of small sizes is in combining these two very different metrics.

The new coreset construction helps to design the first $(1 + \epsilon)$-approximation for center-based clustering problems in UDGs and USGs, that is fixed-parameter tractable in k and ϵ (FPT-AS).

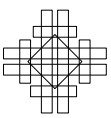
2012 ACM Subject Classification Theory of computation → Computational geometry; Theory of computation → Facility location and clustering; Theory of computation → Sparsification and spanners

Keywords and phrases k-median, k-means, clustering, coresets, geometric graphs

Digital Object Identifier 10.4230/LIPIcs.SoCG.2023.10

Related Version *Full Version*: https://arxiv.org/abs/2303.01400

Funding The research leading to these results has received funding from the Research Council of Norway via the project BWCA (grant no. 314528), and the European Research Council (ERC) via grant LOPPRE, reference 819416.



1 Introduction

Clustering is one of the most important data analysis techniques where the goal is to partition a dataset into a number of groups such that each group contains similar set of data points. The notion of similarity is captured by a distance function between the data points, and the goal of retrieving the best natural clustering of the data points is achieved by minimizing a proxy cost function. In this work, we study the popular (k, z)-clustering problem.

(k, z)-clustering. Given a set of points P in a metric space (Ω, d) and two positive integers k and z, find a set C of k points (or centers) in Ω that minimizes the following cost function:

$$\mathsf{cost}(C) = \sum_{p \in P} \mathsf{cost}(p, C)$$

where $\mathsf{cost}(p, C) = (\mathsf{d}(p, C))^z$ and $\mathsf{d}(p, C) = \min_{c \in C} \mathsf{d}(p, c)$.

Two widely studied clustering problems, k-means, and k-median clustering, are special versions of (k, z)-clustering with $z = 2$ and $z = 1$, respectively. A popular way of dealing with large data for the purpose of the analysis is to apply a data reduction scheme as a preprocessing step. In the context of clustering, one such way of preprocessing the data is to construct an object known as *coresets*.

Coresets. Informally, an ϵ-coreset for (k, z)-clustering is a small-sized summary of the data that approximately (within $(1 \pm \epsilon)$ factor) preserves the cost of clustering with respect to any set of k centers (we will often shorten "ϵ-coreset" to simply "coreset"). Thus, any solution set of centers computed for the coreset points can be readily used as a solution for the original dataset. A formal definition follows.

▶ **Definition 1** (ϵ-Coreset). *A coreset for (k, z)-clustering problem on a set P of points in a metric space (Ω, d) is a weighted subset Y of Ω with weights $\omega : Y \to \mathbb{R}^+$ such that for any set $\mathcal{S} \subseteq \Omega$ with $|\mathcal{S}| = k$,*

$$\left| \sum_{p \in P} \mathsf{cost}(p, \mathcal{S}) - \sum_{p \in Y} \omega(p) \mathsf{cost}(p, \mathcal{S}) \right| \leq \epsilon \cdot \sum_{p \in P} \mathsf{cost}(p, \mathcal{S}).$$

Feldman and Langberg [19] showed that for n points in any general metric, a coreset of size $\mathcal{O}(\epsilon^{-2z} k \log k \log n)$ can be constructed in time $\tilde{\mathcal{O}}(nk)$, where $\tilde{\mathcal{O}}()$ notation hides a poly-logarithmic factor. Also, it is known that the dependency on $\log n$ in the above bound cannot be avoided [3, 17]. However, for several special metrics, it is possible to construct coresets whose size does not depend on the data size. There has been a large pool of work for Euclidean spaces, culminating in a bound of $\mathcal{O}(k \cdot (\log k)^{\mathcal{O}(1)} \cdot 2^{\mathcal{O}(z \log z)} \epsilon^{-2} \cdot \min\{\epsilon^{-z}, k\})$ [17], which is independent of the data size and dimension of the space. Similarly, coresets of size independent of n are also obtained in other specialized settings such as doubling metrics [27], shortest-path metrics in the graphs of bounded-treewidth [3], and graphs excluding a fixed minor [8]. A recent result by Cohen-Addad et al. [17] gives a unified framework that encompasses all these results. We note that, since the number of distinct (weighted) points in coresets is usually much smaller (and sometimes independent of) n, they naturally find applications in non-sequential settings such as streaming [26, 13].

Let us remark that all known results about small coresets in graph metrics strongly exploit the sparse nature of graphs such as bounded treewidth [3] or excluding a fixed minor [8]. There is a very good reason for that. In a complete graph, by setting suitable weights on the edges one can represent *any* general metric. Thus if a graph family contains large cliques, clustering in such graphs is as difficult as in general metrics.

In this work, we are interested in coreset construction for edge-weighted geometric intersection graphs with shortest-path metric. A geometric intersection graph of a set of geometric objects contains a vertex for each object and an edge corresponding to each pair of objects that have non-empty intersection. (We note that for our purpose of designing algorithms, we do not explicitly need the objects or their geometric representation. It is sufficient to work with the graph representation as long as the edge-weights are given.) In particular, geometric intersection graphs are a widely studied model for ad-hoc communication and wireless sensor networks [38, 4, 34, 37, 31]. Notably, clustering is a common topology management method in such networks. Grouping nodes are used as subroutines for executing various tasks in a distributed manner and for resource management, see the survey [39] for an overview of different clustering methods for wireless sensor networks.

Our work is motivated by the following question: *"Is it possible to exploit the properties of geometric intersection graphs for obtaining coresets whose size does not depend on the data size?"* In general, the answer to this question is *no*. This is because geometric intersection graphs can contain large cliques. Even for objects as simple as unit squares, the corresponding intersection graph could be a clique, and, as we already noted, by setting suitable weights on the edges of the clique, one can represent any metric. Hence constructing coresets in geometric intersection graphs with arbitrary edge weights is as difficult as in general metrics. Thus, we need to restrict edge weights in some manner in order to obtain non-trivial coresets for geometric intersection graphs. As an illustrative example, let us take a look at Euclidean-weighted UDGs, a well-studied class of geometric intersection graphs.

Euclidean-weighted unit-disk graph metric. A unit-disk graph (UDG) is defined in the following way – there is a configuration of closed disks of radii 1 in the plane and a one-to-one correspondence between the vertices and the centers of the disks such that there is an edge between two vertices if and only if the disks having the two corresponding centers intersect. The weight of an edge is equal to the Euclidean distance between the two corresponding centers. Euclidean-weighted UDGs have been well-studied in computational geometry [10, 24]. Apart from practical motivation, UDGs are interesting from theoretical perspectives as well. On the one hand, being embedded on the plane they resemble planar graphs when "zoomed out", but could contain large cliques locally. On the other hand, the metric induced by them is an amalgamation of geometric and graphic settings, as it is locally Euclidean but globally a graph metric. Due to the latter property, UDG metric can be used for fine-tuned clustering, as with pure Euclidean distances one can only retrieve clusters induced by convex partitions of the space (see Figure 1).

1.1 Our Results

We now formalize our intuition about the "hybrid" nature about the Euclidean weighted UDGs, by identifying two canonical geometric properties of a graph G that are sufficient for constructing small-sized coresets. For better exposition, we fix a few notations. For any subgraph H of G, we denote its set of vertices and set of edges by $V(H)$ and $E(H)$, respectively. For vertex set $V' \subseteq V$, we denote by $G[V']$ the subgraph of G induced by V'. For a subgraph H of G, and $u, v \in V(H)$, let $\pi_H(u, v)$ denote a shortest path between u and v (according to the edge-weights in G restricted to H) that uses the edges of H, and let $\mathsf{d}_H(u, v)$ denote the weight of $\pi_H(u, v)$, i.e., the sum of the weights of the edges along $\pi_H(u, v)$. For any path π in a graph, let $|\pi|$ denote the number of edges on π. Note that d_H is the so-called *shortest path metric* on H. Finally, for any pair of points $p, q \in \mathbb{R}^2$, let $|pq|$ denote the euclidean (i.e., ℓ_2-norm) distance between p and q.

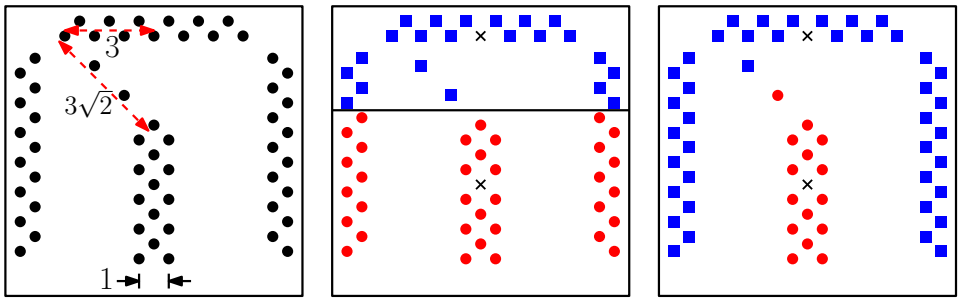


Figure 1 (Left.) A point set in 2D. (Middle.) 2-means clustering with Euclidean distances. Centers are shown by crosses. Points of two clusters are shown by disks (red) and squares (blue). (Right.) 2-means clustering on UDG – due to the $3\sqrt{2}$ diagonal path distance, all the blue points (squares) are closer to the upper center.

Canonical geometric properties.

(1) *Locally Euclidean:* There exist (not necessarily distinct) constants $c_1, c_2, c_3, c_4 \geq 0$, such that the following holds. G has an embedding $\lambda : V(G) \to \mathbb{R}^2$ in the plane such that the vertices of G are mapped to points in the plane, with the following two properties.

1. For any two $u, v \in V(G)$, if $|\lambda(u)\lambda(v)| \leq c_1$ then $uv \in E(G)$, and for any $u', v' \in V(G)$, if $|\lambda(u')\lambda(v')| > c_2$, then $u'v' \notin E(G)$.
2. For any $u, v \in V(G)$ such that $uv \in E(G)$, let $w(uv)$ denote the weight of the edge uv. Then, the edge uv is a shortest path between u and v in G.
   Furthermore, $c_3 \cdot |\lambda(u)\lambda(v)| \leq w(uv) \leq c_4 \cdot |\lambda(u)\lambda(v)|$.

(2) *Planar Spanner:* For any induced subgraph $G' = G[V']$ with $V' \subseteq V(G)$, there exists a planar α-spanner $H' = (V', E(H'))$ for some fixed $\alpha \geq 1$, i.e., (i) H' is a subgraph of G' (and hence of G) – $E(H') \subseteq E(G')$, and (ii) for any $u, v \in V'$, $\mathsf{d}_{G'}(u, v) \leq \mathsf{d}_{H'}(u, v) \leq \alpha \cdot \mathsf{d}_{G'}(u, v)$.

Our main result is the following theorem.

▶ **Theorem 2** (Informal). *Consider the metric space (V, d_G) induced by any graph G satisfying the two canonical geometric properties (1) and (2), and a set $P \subseteq V(G)$. Then there exists a polynomial time algorithm that constructs a coreset for (k, z)-clustering on P of size $\mathcal{O}(\epsilon^{-\beta} k \log^2 k)$, where $\beta = \mathcal{O}(z \log z)$.*

Theorem 2 is a handy tool to construct coresets for several interesting geometric intersection graphs coupled with suitable metrics. First, let us observe that our initial example, namely, a metric induced by a Euclidean-weighted UDG G satisfies the two canonical properties. Consider an embedding of G in $\mathbb{R}^2$. Note that there is an edge between any two points iff the Euclidean distance between the two points is at most 2, and the weight of such an edge is exactly the euclidean distance. Thus, G is *Locally Euclidean* with $c_1 = c_2 = 2$, and $c_3 = c_4 = 1$. Furthermore, due to a result of Li, Calinescu, and Wan [36], any Euclidean-weighted UDG admits a constant-stretch planar spanner. Thus, G also satisfies the *Planar Spanner* property. Therefore, due to Theorem 2, we can obtain $\mathcal{O}(\epsilon^{-\beta} k \log^2 k)$-size coresets for (k, z)-clustering on Euclidean-weighted UDGs. In the following, we discuss further applications of our framework.

ℓ_∞-weighted unit-square graph metric. Unit-square graphs (USGs) are similar to UDGs except they are defined as intersection graphs of (axis-parallel) unit squares instead of unit disks [1]. Indeed, these two graph classes are distinct. For example, the $K_{1,5}$ claw can be realized by a UDG, but not by any USG. (See Figure 2.) Since a unit square is a "unit ball" in ℓ_∞-norm, it is more natural to consider ℓ_∞ weights on the edges. It is not too difficult to see that the *Locally Euclidean* property holds for ℓ_∞-weighted USGs – we give a formal proof in the full version given in the appendix. On the other hand, in order to establish the second property, we have to prove the existence of a constant-stretch planar spanner for USGs. To the best of our knowledge this result was previously not known and is of independent interest. We give a proof of this result in the full version. Thus, ℓ_∞ weighted USGs also satisfy the two properties required to apply Theorem 2 in order to obtain a small-sized coreset.

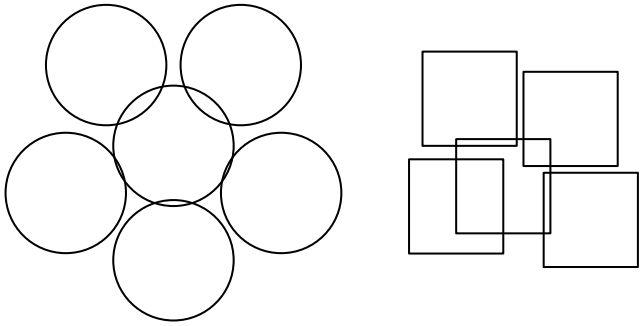


Figure 2 A set of disks realizing $K_{1,5}$ (left) and a set of squares realizing $K_{1,4}$ (right). A pair of intersecting unit squares must contain a corner of each other, and so the central square can intersect with at most four other unit squares that are pairwise disjoint.

Other extensions. In $\mathbb{R}^2$, all ℓ_p distances ($1 \leq p \leq \infty$) are within a $\sqrt{2}$ factor from each other. Thus, our arguments can be easily extended to any ℓ_p weights on UDGs/USGs for any $1 \leq p \leq \infty$ without any changes on the bounds (a formal argument can be found in the full version). Lastly, we consider shortest-path metrics in unweighted (i.e., hop-distance) unit-disk graphs of bounded degree. Notably, these graphs satisfy the *Planar Spanner* property due to a result by [7], but not the *Locally Euclidean* property. Nevertheless, we can modify our approach to construct a small-sized coreset for such metrics. To summarize, we obtain coresets for (k, z)-clustering with size independent of n for the following graph metrics.

- ℓ_p-distance weighted UDGs for any $1 \leq p \leq \infty$,
- ℓ_p-distance weighted USGs for any $1 \leq p \leq \infty$,
- Bounded-degree unweighted UDGs.

FPT Approximation Schemes. As a corollary to Theorem 2, we obtain $(1 + \epsilon)$-approximations for (k, z)-clustering in geometric intersection graphs that are fixed-parameter tractable (FPT) in k and ϵ. Note that such a $(1 + \epsilon)$-approximation was not known before even for UDGs, as it does not follow from previously known bound on coreset sizes. Prior to our work, the best known bound for coresets on UDGs – as in general metrics – was $O(k \log n \cdot \epsilon^{-\max(2,z)})$ [18]. Even though a coreset reduces the number of distinct points

[1] Although it might seem unnatural at first, it is convenient to define a *unit square* as a square of sidelength 2. This is analogous to a unit disk being a disk of diameter 2. In either case, the class of USGs remains unaffected by scaling.

(or clients) to be clustered, the number of potential centers (or facilities) still remains the same, i.e., n. Hence, a coreset does not directly help us enumerate all possible sets of k centers from which we could pick the best set. An alternative way to enumerate these sets of centers is to enumerate all possible partitions (or clusterings) of the coreset points. Note that each clustering of coreset points corresponds to a clustering of the original points, and the cost of clustering is preserved to within a $(1 \pm \epsilon)$ factor. With our coreset bound of $\mathcal{O}(\epsilon^{-\beta} k \log^2 k)$, the number of distinct clusterings is only $k^{\mathcal{O}(\epsilon^{-\beta} k \log^2 k)}$. Overall the algorithm takes $k^{\mathcal{O}(\epsilon^{-\beta} k \log^2 k)} n^{\mathcal{O}(1)}$ time.

▶ **Corollary 3.** *For each of the metrics listed in the above, there exists a $(1+\epsilon)$-approximation for (k, z)-clustering with $z \geq 1$ that runs in time $2^{\mathcal{O}(\epsilon^{-\beta} k \log^3 k)} n^{\mathcal{O}(1)}$, where $\beta = \mathcal{O}(z \log z)$.*

1.2 General overview of the methods

Our coreset construction is based on a recent work due to Cohen-Addad, Saulpic, and Schwiegelshohn [18], which gives a framework for constructing coresets in various settings.The essence of the framework is that it translates the problem of coreset construction to showing the existence of an interesting set of centers or centroid set. In particular, consider any set $\mathcal{S}$ of k centers and any subset $X \subseteq P$ of points that are sufficiently close to $\mathcal{S}$ compared to an existing solution $\mathcal{A}$. Then a subset $\mathbb{C} \subseteq \Omega$ is a *centroid set* for P if it contains centers that well-approximates $\mathcal{S}$, i.e., there exists $\tilde{\mathcal{S}} \subseteq \mathbb{C}$, such that for every $p \in X$, it holds that $|\mathsf{cost}(p, \mathcal{S}) - \mathsf{cost}(p, \tilde{\mathcal{S}})| \leq \epsilon(\mathsf{cost}(p, \mathcal{S}) + \mathsf{cost}(p, \mathcal{A}))$. The framework informally states that if there is a centroid set $\mathbb{C}$, then a coreset can be constructed whose size depends logarithmically on $|\mathbb{C}|$. Such a dependency arises in their randomized construction in order to prove a union bound over all possible interesting solutions, which can be at most $|\mathbb{C}|^k$. By showing the existence of small-sized centroid sets, they obtain improved coreset size bounds for a wide range of spaces.

We use the framework of Cohen-Addad et al. for our coreset construction. Our main technical result shows if for a graph G with metric d_G, the two canonical geometric properties are satisfied, then there exists a small-sized centroid set for G. This is the most challenging part of the proof and it requires a novel combination of tools and techniques from computational geometry. As soon as we establish the existence of the centroid set, the construction of coresets follows the steps of [18]. For the sake of exposition, let us consider a concrete example of Euclidean-weighted UDGs.

Consider any cluster of points with cluster center s. The points that are nearby (i.e., within distance 2) s behave simply as points in the Euclidean case. But, a point p that is far away from s can have a shortest path distance which may be much larger than the actual Euclidean distance between p and s, see Figure 1. We first show that it is possible to conceptually separate out these two cases – but one has to be careful, as a cluster can potentially contain both types of points. Notably, none of the previous works had to deal with such a hybrid metric. To handle the set of nearby points, we exploit the *Locally Euclidean* property. In particular, by overlaying a grid of appropriately small sidelength, and selecting one representative point from each cell of the grid, we can compute a centroid set that preserves the distances from the nearby points.

In the other case, a shortest path between a point p and a center s consists of more than one edge, and we need to deal with a graphical metric. This case is much more interesting. All other works establishing small-sized centroid sets in certain graph metrics exploit the fact that certain graph classes admit small or well-behaved separators. For example, bounded treewidth graph admit separators bounded by treewidth; whereas graphs excluding a fixed

minor admit shortest path separators. However, UDGs may contain arbitrarily large cliques, and therefore do not admit such separators in general. Thus, we reach a technical bottleneck. Note that this is the first work of its kind that handles such a dense graph. To overcome this challenge, we use the other canonical property. Instead of directly working with the UDG, we consider its planar spanner, where distances are preserved up to a constant factor. The existence of such a spanner is guaranteed by the second canonical property, *Planar Spanner*. As planar graphs have shortest path separators, now we can apply the existing techniques. However, if we were to entirely rely on the spanner, some of the distances may be scaled up by a constant (> 2) factor in the spanner, and thus it would not be possible to ensure the $(1 \pm \epsilon)$-factor bound required to construct a coreset.

Thus, we use the spanner as a supporting graph in the following way. First, we recursively decompose the original UDG by making use of the shortest path separators admitted by the planar spanner. We note that although planar graph decomposition has been used in coreset literature, using such a guided scheme to obtain a decomposition of a much more general graph is novel. Then, we use this recursive decomposition of the UDG, along with the shortest path separators used to find this decomposition, in order to construct the centroid set. In this construction, we use the spanner in a restricted manner, and use it such that error incurred by the use of the spanner is upper bounded by α times the weight of at most one edge along a shortest path from a point p to its corresponding (approximate) center. However, observe that if such a shortest path consists of a single edge, then even this error is too large. To resolve this issue, we rely on the planar spanner, only if the shortest path is "long enough", i.e., contains $\Omega(z/\epsilon)$ edges. In this case, *Locally Euclidean* property implies that for such a "long path", the *length* of the path and the *number of edges* on the path are within a constant factor from each other. This implies that the error introduced by rerouting a single edge using the spanner is at most ϵ times the length of the path, i.e., negligible.

Finally, if a shortest path between a point and a center consists of $\mathcal{O}(z/\epsilon)$ edges, then we can use a modified version of the grid-cell argument to obtain a small-sized centroid set.

We note that this is simply an intuitive overview of the challenges faced in each of the three cases. The actual construction of the centroid set, and the analysis of the error incurred in each of the cases is fairly convoluted. While replacing a center $s \in \mathcal{S}$ by another one $\tilde{s} \in \mathbb{C}$, we need to ensure that for a point p having s as its closest center, $\mathsf{d}_G(p, \tilde{s})$ is neither too large nor too small compared to $\mathsf{d}_G(p, s)$, since we want to bound the error in the absolute difference. In addition, we have to be extremely careful while combining the three centroid (sub)sets constructed for each of the cases, and ensure that a good replacement $\tilde{s}$ found for a center s in one of the cases does not adversely distort the error for a point that is being handled in another case.

Related work. Here we give an overview of the literature on coresets. For a more exhaustive list, we refer to [18, 28]. Coreset construction was popularized by an initial set of works that obtained small-sized coresets in low-dimensional Euclidean spaces [26, 25, 22]. Chen [13] obtained the first coreset for Euclidean spaces with polynomial dependence on the dimension and the first coreset in general metrics, where the size is $\mathcal{O}(k^2\epsilon^{-2}\log^2 n)$ for k-median. Subsequently, the dependence on the dimension has been further improved [33, 20]. Finally, such dependence was removed in [21, 40]. See also [5, 18, 28, 8, 17] for recent improvements.

Both k-median and k-means admit polynomial-time $\mathcal{O}(1)$-approximations in general metrics [11, 12, 29, 35, 9, 1]. Moreover, algorithms with improved approximation guarantees can be obtained that is FPT in k and ϵ [15]. Naturally, the problems have also been studied in specialized metrics. Polynomial-time approximation schemes (PTASes) are known for

Euclidean k-median [2] and k-means [16, 23]. See [14, 32] for other improvements. Similar to geometric clustering, clustering in graphic setting is also widely studied. PTASes are known for excluded-minor graphs [16, 8].

Also, FPT approximation schemes are known for graphs of bounded-treewidth [3] and graphs of bounded highway dimension [6, 8].

2 Coresets for Geometric Graphs

To set up the stage, we need the following definition of *centroid set* from [18].

▶ **Definition** (Centroid Set). *Consider any metric space (Ω, d), a set of clients $P \subseteq \Omega$, and two positive integers k and z. Let $\epsilon > 0$ be a precision parameter. Given a set of centers $\mathcal{A}$, a set $\mathbb{C}$ is an $\mathcal{A}$-approximate centroid set for (k, z)-clustering on P that satisfies the following property.*

For every set of k centers $\mathcal{S} \subseteq \Omega$, there exists $\tilde{\mathcal{S}} \subseteq \mathbb{C}$, such that for every $p \in P$ that satisfies $\mathsf{cost}(p, \mathcal{S}) \leq \left(\frac{10z}{\epsilon}\right)^z \cdot \mathsf{cost}(p, \mathcal{A})$ or $\mathsf{cost}(p, \tilde{\mathcal{S}}) \leq \left(\frac{10z}{\epsilon}\right)^z \cdot \mathsf{cost}(p, \mathcal{A})$, it holds that

$$|\mathsf{cost}(p, \mathcal{S}) - \mathsf{cost}(p, \tilde{\mathcal{S}})| \leq \frac{\epsilon}{z \log(z/\epsilon)} (\mathsf{cost}(p, \mathcal{S}) + \mathsf{cost}(p, \mathcal{A})).$$

Informally, a centroid set $\mathbb{C}$ is a collection of candidate centers, potentially much smaller than Ω, such that the k centers in $\mathcal{S}$ can be replaced by k centers in $\tilde{\mathcal{S}} \subseteq \mathbb{C}$ without changing the cost of points by a large amount, that are much closer to $\mathcal{S}$ or $\tilde{\mathcal{S}}$ compared to $\mathcal{A}$ w.r.t. d. Cohen-Addad et al. [17] proved that one can obtain coresets whose size depends only logarithmically on the size of any such centroid set. More formally, they prove the following.

▶ **Proposition 4** ([18]). *Consider any metric space (Ω, d), a set of points $P \subseteq \Omega$ with n distinct points, and two positive integers k and z. Let $\epsilon > 0$ be a precision parameter. Suppose $\mathcal{A}$ be a given constant-factor approximation for (k, z)-clustering on P.*

Suppose there exists an $\mathcal{A}$-approximate centroid set for (k, z)-clustering on P. Then there exists a polynomial time algorithm that constructs with probability at least $1 - \delta$ a coreset of size

$$\mathcal{O}\left(\frac{2^{\mathcal{O}(z \log z)} \cdot \log^4(1/\epsilon)}{\min\{\epsilon^2, \epsilon^z\}} (k \log |\mathbb{C}| + \log\log(1/\epsilon) + \log(1/\delta))\right)$$

with positive weights for (k, z)-clustering on P.

First, note that the above coreset framework requires only existence of such a centroid set. It is not necessary to explicitly compute it. Indeed, such a centroid set is only used to bound the size of computed coresets in their analysis. The main contribution of our work is to obtain small-sized centroid sets for geometric graph metrics that satisfy the two canonical geometric properties. In particular, we prove the following theorem.

▶ **Theorem 5** (Centroid Set Theorem). *Consider the metric space (V, d_G) induced by any graph $G = (V, E)$ satisfying the Locally Euclidean and Planar Spanner properties defined before. Also consider a set of points $X \subseteq V$ and two positive integers k and $z \geq 1$. Let $\epsilon > 0$ be the precision parameter. Additionally, suppose $\mathcal{A}$ be a solution for (k, z)-clustering on X. Then there exists an $\mathcal{A}$-approximate centroid set $\mathbb{C}$ for (k, z)-clustering on X of size $\exp(\mathcal{O}(\log^2 |X| + z^{16}\epsilon^{-8}(\log(z/\epsilon))^8 \log |X|))$.*

We give an overview of the proof of Theorem 5 in the following section, and defer a formal proof to the Section 3 of the full version. Then, by combining Proposition 4 and arguments from [8], with some minor changes due to our different bound on coreset-size, we obtain the following theorem. A formal proof can be found in Section 2 of the full version.

▶ **Theorem 6.** *Consider the metric space (V, d_G) induced by any graph $G = (V, E)$ satisfying Locally Euclidean and Planar Spanner properties, a set $P \subseteq V$ with n distinct points, and two positive integers k and $z \geq 1$. Then there exists a polynomial time algorithm that constructs with probability at least $1 - \delta$ a coreset for (k, z)-clustering on P of size $\mathcal{O}(\epsilon^{-\mathcal{O}(z \log z)} k \log^2 k \log^3(1/\delta))$, where z is a constant, and $\delta < 1/4$.*

3 Overview of the Proof of Centroid Set Theorem

In this section, we give an overview of our main result, namely, the existence of a small-sized centroid set. A formal proof can be found in the full version.

Recall that we are given $G = (V, E)$, a connected, undirected, and edge-weighted graph on n vertices. Moreover, G satisfies the two canonical geometric properties, namely *Locally Euclidean*, and *Planar Spanner*.

As (V, d_G) is our metric space, we use the terms points and vertices interchangeably. $X \subseteq V$ is the given set of points. We are also given $\mathcal{A}$, a solution for (k, z)-clustering on X. We prove that there exists an $\mathcal{A}$-approximate centroid set of size $\exp(\mathcal{O}(\log^2 |X| + z^{16}\epsilon^{-8}(\log(z/\epsilon))^8 \log |X|))$ for (k, z)-clustering on X, which satisfies the following property.

> For every set of k centers $\mathcal{S} \subseteq V$, there exists $\tilde{\mathcal{S}} \subseteq \mathbb{C}$, such that for every $p \in X$ that satisfies $\mathsf{cost}(p, \mathcal{S}) \leq \left(\frac{10z}{\epsilon}\right)^z \cdot \mathsf{cost}(p, \mathcal{A})$ or $\mathsf{cost}(p, \tilde{\mathcal{S}}) \leq \left(\frac{10z}{\epsilon}\right)^z \cdot \mathsf{cost}(p, \mathcal{A})$, it holds that
>
> $$|\mathsf{cost}(p, \mathcal{S}) - \mathsf{cost}(p, \tilde{\mathcal{S}})| \leq \frac{\epsilon}{z \log(z/\epsilon)} (\mathsf{cost}(p, \mathcal{S}) + \mathsf{cost}(p, \mathcal{A})).$$

Now we proceed to an overview of the proof of the theorem. This proof can be divided into three steps.

1. Construction of a centroid set $\mathbb{C}$ of a small size.
2. Given a solution $\mathcal{S} \in V^k$, finding for each center $s \in \mathcal{S}$, a *replacement* center $\rho(s) \in \mathbb{C}$, to construct $\tilde{\mathcal{S}} \in \mathbb{C}^k$.
3. Showing that $\tilde{\mathcal{S}}$ approximates $\mathcal{S}$, i.e., it satisfies the property specified above.

Given the hybrid nature of the metric, each of these three steps is subdivided into multiple cases. Recall that, due to the first canonical property, points in V that are close to each other behave as in the Euclidean case. To take care of the case of **points nearby** to their closest centers, we add a set of points $\mathbb{C}_{\text{net}}$ to our centroid set. The case of far away points is further divided into two subcases. In the first subcase, we deal with the points whose shortest paths to closest centers are **short** or $\mathcal{O}(z/\epsilon)$ hops away. To take care of this subcase, we add a set of points $\mathbb{C}_{\text{support}}$ to our centroid set. The last subcase concerns **long paths**, and here we make use of the planar spanner property. In particular, we construct the centroid points in this case based on a recursive decomposition of the graph guided by underlying planar spanners of the decomposed subgraphs. This subcase resembles the centroid set construction in excluded-minor graph metrics from [17, 8]. In this informal overview, it will be convenient to consider each of these three cases separately, and discuss steps 1-3 in each case (overview of step 1 in each case hints at why the size of centroid set is bounded, but a formal proof is given in the full version).

Nearby points case.

Construction of $\mathbb{C}_{\text{net}}$. Let $p_1, p_2, \ldots, p_{n'}$ be the points of X such that $\mathsf{d}_G(p, \mathcal{A}) < 1$ for all $p = p_i$, where $1 \leq i \leq n'$. Let $B_i = B(p_i, (10z/\epsilon) \cdot \mathsf{d}_G(p_i, \mathcal{A}))$ be the Euclidean ball

centered at p_i having radius $(10z/\epsilon) \cdot \mathsf{d}_G(p_i, \mathcal{A})$. For each $1 \leq i \leq n'$, compute an $(\epsilon^3/z^3) \cdot \mathsf{d}_G(p_i, \mathcal{A})$-net of the disk B_i that is a subset of V and add that to $\mathbb{C}_{\text{net}}$. That is, this net picks at most one point of V from each gridcell of length $\beta = (\epsilon^3/z^3) \cdot \mathsf{d}_G(p_i, \mathcal{A})$ that intersects with the disk B_i. Note that the distance between any two points that belong to the same cell is $\mathcal{O}(\beta)$. In particular, the *Locally Euclidean* nature of G helps us define this net.

Finding a replacement center. For a center $s \in \mathcal{S}$, let $X_s \subseteq X$ denote the set of points that have s as their closest center in $\mathcal{S}$. If there exists a $p_i \in X_s$ with $\mathsf{d}_G(p, \mathcal{A}) < 1$, then we argue that s belongs to the ball B_i as defined above. It follows that we added a point $\tilde{s}$ to $\mathbb{C}_{\text{net}}$, such that $\mathsf{d}_G(s, \tilde{s}) = \mathcal{O}(\frac{\epsilon^3}{z^3}\mathsf{d}_G(p_i, \mathcal{A}))$. We pick $\tilde{s}$ as the replacement for s, i.e., $\rho(s) = \tilde{s}$, and add it to $\tilde{\mathcal{S}}$.

Error analysis. Using triangle inequality, it follows that for any $p \in X_s$, $|\mathsf{d}_G(p, s) - \mathsf{d}_G(p, \tilde{s})| = \mathcal{O}(\frac{\epsilon^3}{z^3}\mathsf{d}_G(p_i, \mathcal{A}))$. If there are multiple choices for p_i, a careful choice ensures that this error is upper bounded by $\epsilon/z \cdot (\mathsf{d}_G(p, s) + \mathsf{d}_G(p, \mathcal{A}))$.

Short-path case.

Construction of $\mathbb{C}_{\text{support}}$. For each point $p \in X$, we consider a disk $D(p, r)$ of radius $r = \Theta(z/\epsilon)$ around p. Finally, we consider the union of the area covered by all such disks. We subdivide this area into small gridcells of sidelength $\mathcal{O}(\epsilon^2/z^2)$, and select one representative point from each grid cell, and add it to the set $\mathbb{C}_{\text{support}}$. [2]

Finding a replacement center. For a center $s \in \mathcal{S}$, suppose we were not able to find a replacement using the previous case. Then, if $\tilde{s}$ from the same cell as s of sidelength $\mathcal{O}(\epsilon^2/z^2)$ was added to $\mathbb{C}_{\text{support}}$, then we set $\rho(s) = \tilde{s}$, and add it to $\tilde{\mathcal{S}}$.

Error analysis. If a shortest path $\pi_G(p, s)$ is *short*, i.e., contains at most ℓ edges, where $\ell = \Theta(z/\epsilon)$, then due to the *Locally Euclidean* property, s belongs to the disk $D(p, r)$. It follows that we select some $\tilde{s}$ in $\mathbb{C}_{\text{support}}$ such that $\mathsf{d}_G(s, \tilde{s}) = \mathcal{O}(\epsilon^2/z^2)$. Again, using triangle inequality, it follows that for any $p \in X_s$, it holds that $|\mathsf{d}_G(p, s) - \mathsf{d}_G(p, \tilde{s})| = \mathcal{O}(\epsilon^2/z^2) \leq \mathcal{O}(\epsilon^2/z^2) \cdot \mathsf{d}_G(p, \mathcal{A})$. Here, the second inequality follows since for any point $p \in X_s$, $\mathsf{d}_G(p, \mathcal{A}) > 1$; otherwise the previous case would apply. Note that in this simplified argument, precision value of ϵ/z would also suffice; however, in the actual analysis we need the more granular value of ϵ^2/z^2 to ensure that the choices made in **nearby points** and **short path** cases do not affect each other.

Long-path case. This is the most involved case out of the three, and it is here that we rely on the *planar spanner* property. Before going into the details, let us first note the following property. For any $s \in \mathcal{S}$ whose replacement has not been found in the previous two cases, it holds that *for every* $p \in X$, $|\pi(p, s)| > \ell$. That is, the hop-distance of s from every point in X is strictly larger than ℓ. This observation will be crucial in the subsequent error analysis.

Construction of $\mathbb{C}_{\text{landmark}}$. Here, we first use the *Planar Spanner* property to obtain a recursive decomposition of the graph represented by a tree $\mathcal{T}$ (see Figure 3 for an illustration). Each node of $\mathcal{T}$ corresponds to a subset of vertices of G (called *region*). By slightly abusing the notation, we equate a node i with its corresponding region $R_i \subseteq V(G)$. The decomposition ensures that the union of all children regions of R_i is equal to R_i. This decomposition is obtained as follows. The root region R_1 is equal to $V(G)$. Now, consider a region R_i, and the corresponding induced subgraph $G_i = G[R_i]$. Due to *Planar*

[2] In the actual construction, we define a *support graph* to define the set $\mathbb{C}_{\text{support}}$. But, at a high level, the construction follows the overview given here.

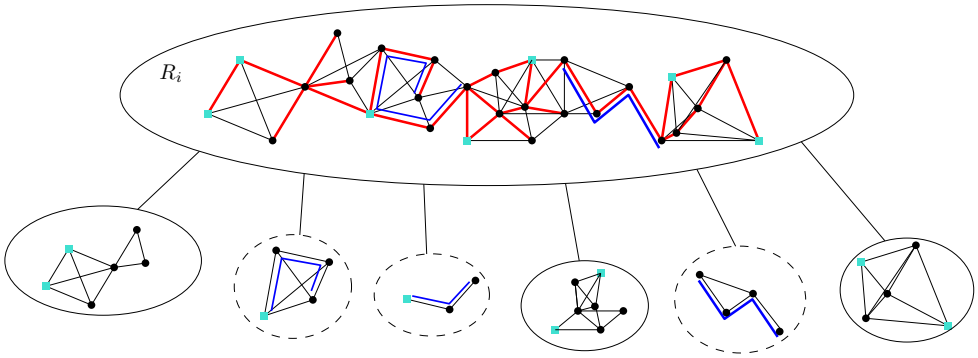


Figure 3 Decomposition of a region R_i into multiple children. Inside R_i, we show the induced subgraph $G_i = G[R_i]$ (which is not planar) and the corresponding planar spanner H_i. The edges of H_i are shown in red, and the edges in $E(G_i) \setminus E(H_i)$ are in black. The two shortest-path separators of H_i are shown in blue, which form a balanced separator $\mathcal{P}_i$ for the vertices of X (shown as light blue squares). Then, we add a child for every connected component of $H \setminus V(\mathcal{P}_i)$ (children regions inside solid ellipses). We also have children corresponding to paths in $\mathcal{P}_i$ (dashed ellipses). One of the paths is broken into two pieces due to a vertex in X.

Spanner property, G_i admits a constant stretch planar spanner H_i. We then use the following well-known result.

▶ **Proposition 7** ([10, 30]). *Given an edge-weighted planar graph H', with non-negative weights on vertices, there exists a collection of shortest paths $\mathcal{P} = \{P_1, P_2, \ldots, , P_b\}$ with $b = \mathcal{O}(1)$, such that the set of vertices of every connected component in $H \setminus \bigcup_{Pi \in \mathcal{P}} V(P_i)$ has weight at most half of that of $V(H')$.*

We define the weight of a vertex in H_i as 1 if it belongs to X, and 0 otherwise. Then, by applying Proposition 7, we obtain a collection of shortest paths $\mathcal{P}_i$ in H_i such that each connected component of $H_i \setminus V(\mathcal{P}_i)$ contains at most a constant fraction of vertices of X. Now we add children to R_i in the tree $\mathcal{T}$, as follows: there is one child of R_i for a subset of R_i corresponding to (1) each connected component in $H_i \setminus V(\mathcal{P}_i)$, and (2) each shortest path $P^{i,j} \in \mathcal{P}_i$ (if a path $P^{i,j}$ contains vertices of X, then we break it at each such vertex and add the pieces as multiple children). We recursively subdivide the current R_i into multiple children in this manner as long as $|R_i \cap X| \geq 2$. Thus the height of the tree is $\mathcal{O}(\log |X|)$.

Now, consider a leaf node corresponding to a region R_t, and the corresponding root-leaf path $\Pi = (R_1, R_2, \ldots, R_t)$, where R_1 is the root region. For each region R_i along the path, we have a set of $\mathcal{O}(1)$ shortest paths that are separators for the corresponding spanner H_i of $G_i = G[R_i]$. Thus, in total we have a collection of $\mathcal{O}(\log |X|)$ paths corresponding to a region R_t. Now, we select a set of vertices on each of these paths that are at evenly spaced distances, and add these vertices to a set $\mathcal{L}$ of *landmarks*. We also add the vertices in $R_t \cap X$ to the landmark set $\mathcal{L}$. It can be shown that the overall size of $\mathcal{L}$ is $\mathcal{O}_{\epsilon,z}(\log |X|)$. Now, for every vertex $s \in R_t$, we look at the distance vector obtained by looking at distances of s from each landmark in $\mathcal{L}$, and discretize each distance entry by rounding it. Thus, we can partition R_t into a bounded number of equivalence classes based on the *discretized distance vectors* to $\mathcal{L}$. For each leaf R_t, and each equivalence class of R_t, we add one representative point in $\mathbb{C}_{\text{landmark}}$. [3]

[3] In the actual construction, we consider different spacings along each path $P^{i,j}$, which results in multiple

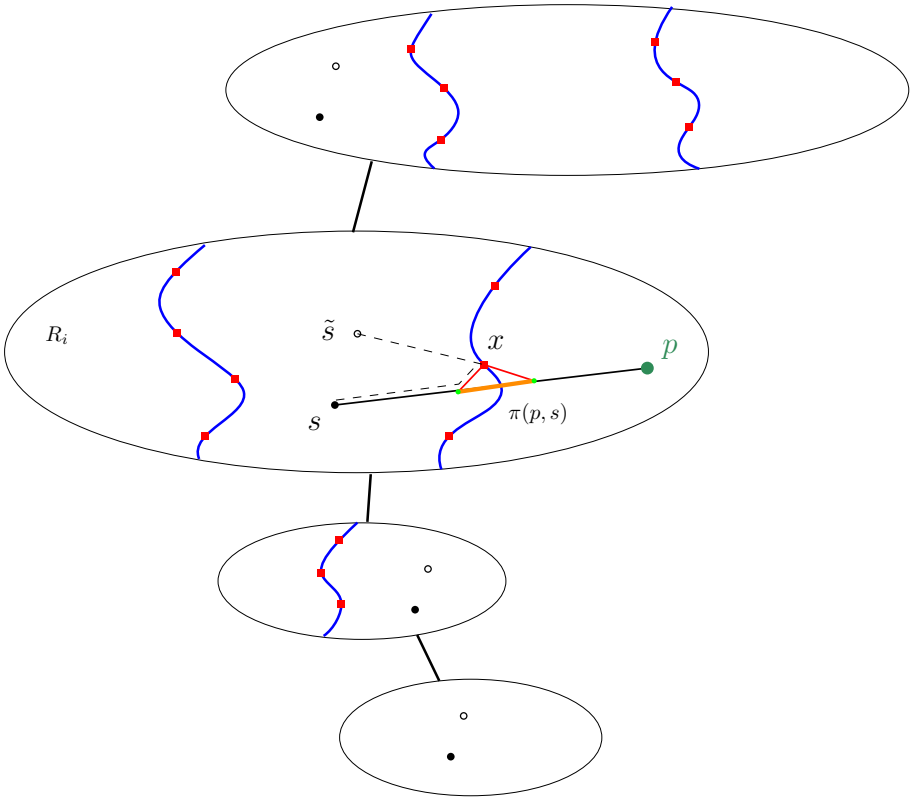


Figure 4 Using landmarks to reroute paths. Here, we show a root-leaf path in $\mathcal{T}$, such that each region along the path contains s and its replacement $\tilde{s}$. This $\tilde{s}$ is chosen such that distances of s and $\tilde{s}$ has same discretized distances w.r.t. all landmarks in $\mathcal{L}$ (shown as red squares), which are spaced evenly along the shortest path separators (shown in blue). Suppose R_i is the lowest region containing s and a point p, which means that a separator path in $\mathcal{P}_i$ separates them. We reroute a single edge (shown in orange) on the shortest path $\pi(p,s)$ via a landmark x (rerouted path shown in red) using the edges of H_i, to obtain an approximate shortest path $\tilde{\pi}(p,s)$. Since s and $\tilde{s}$ have approximately the same distance to x (dashed black paths), we can use this to show that $\mathsf{d}_G(p,s) \approx \mathsf{d}_G(p,\tilde{s})$.

Finding a replacement center. Consider a center $s \in \mathcal{S}$, and suppose it belongs to a leaf region R_t. By construction, we added a point $\tilde{s} \in R_t$ to $\mathbb{C}_{\text{landmark}}$, such that $\tilde{s}$ and s have same discretized distance vector w.r.t. all landmarks in $\mathcal{L}$. We set $\rho(s) = \tilde{s}$, and add it to $\tilde{\mathcal{S}}$.

Error analysis. Now consider a point $p \in X_s$. If $p \in R_t$, then $p \in \mathcal{L}$. Therefore, s and $\tilde{s}$ have same discretized distances w.r.t. p. This ensures that $\mathsf{d}_G(p,s)$ and $\mathsf{d}_G(p,\tilde{s})$ are within the required error bound.

Otherwise, $p \notin R_t$. This means that during the decomposition process, s and p must have been separated when we split a region $R_i \in \Pi$ into its multiple children. If $G_i = G[R_i]$ were planar, then we could conclude that the shortest path $\pi(p,s)$ must intersect some

collections of landmarks for each path. We then consider all possible choices of spacings for each of the $\mathcal{O}(\log|X|)$ paths $P^{i,j}$, which results in multiple landmark sets corresponding to each set of choices. For each landmark set thus obtained, we then define the equivalence classes of R_t in the manner described above, and add one representative of each class to the set $\mathbb{C}_{\text{landmark}}$. However, for the current overview, let us continue with this simplified (although inaccurate) construction.

shortest-path separator $P^{i,j} \in \mathcal{P}_i$. Unfortunately, in our case, G_i is not planar, and $P^{i,j}$ is not a separator for G_i, but for its planar spanner H_i. Here we recall the property that the hop-length of the shortest path $|\pi(p,s)|$ is strictly larger than ℓ, i.e., is $\Omega(z/\epsilon)$, which implies that the weight of a single edge is negligible as compared to the total length of the path. Then, we show that it is possible to reroute a single edge of the path $\pi(p,s)$ using the spanner H_i (which results in a constant factor increase in the distance, but *only for a single edge*) to obtain an *approximately shortest path* $\tilde{\pi}(p,s)$. Furthermore, $\tilde{\pi}(p,s)$ intersects a separator path $P^{i,j}$, and the intersecting vertex x is a landmark. This is the most intricate part of the argument, since we have to argue about shortest paths in G, G_i and H_i. Once we construct $\tilde{\pi}(p,s)$, we can use a subpath of $\tilde{\pi}(p,s)$ to go from p to x, and then use the fact that $\mathsf{d}_G(s,x) \approx \mathsf{d}_G(\tilde{s},x)$, since x is a landmark, and s and $\tilde{s}$ have same discretized distances w.r.t. x. Thus, we can show that $\mathsf{d}_G(p,s)$ and $\mathsf{d}_G(p,\tilde{s})$ are within the required bound. See Figure 4 for an illustration.

Note that the novelty of our work compared to previous works (Braverman et al. [8] and Cohen-Addad et al. [17]) lies in the ability of utilizing an underlying planar spanner instead of the original graph and still achieve a similar error bound sufficient for the analysis.

4 Conclusion

We obtain the first coresets for k-clustering problems whose size is independent of n, on a variety of geometric graph metrics, such as weighted intersection graphs of unit disks and squares. A UDG (or a USG) can contain arbitrarily large cliques, i.e., they can be (locally) dense. Therefore, to the best of our knowledge, ours is the first small-sized (i.e., independent of n) coreset construction for a shortest-path metric on a dense family of graphs. Due to the inherently "hybrid" nature of such metrics, our coreset construction has to carefully navigate the locally-Euclidean and globally-sparse nature of the metric.

We believe the contribution of our work is also conceptual, in that we "abstract out" the geometric properties of such metrics that are sufficient to obtain small-sized coresets via the versatile framework of Cohen-Addad et al. [18]. These structural properties are also satisfied by ℓ_p-norm weighted UDGs and USGs. Furthermore, by suitably modifying the construction, we can also handle hop metrics (i.e., unweighted edges) induced by UDGs of bounded degree. Thus, we obtain small-sized coresets, and thus FPT-approximation schemes, for k-clustering problems for all of these graph families. In order to obtain the result on USGs, we prove that these graphs admit a 3-stretch planar spanner, a result that may be of independent interest.

The most natural question is to find more graph families that satisfy the geometric properties (or some suitable modifications thereof) identified in this work. Disk graphs in $\mathbb{R}^2$ and Unit Ball Graphs in $\mathbb{R}^d$ (for constant $d \geq 3$) are two orthogonal generalizations of UDGs, and thus may be the most obvious candidates. However, these graphs are not known to admit a constant stretch planar spanner. As an intermediate step, it might be interesting to consider "unit disk graphs" that are embeddded on a surface Σ of bounded genus. Here, it might be more natural to require that such a graph admit constant stretch spanner that is also embeddable on Σ (which is a relaxation of planarity). It might be possible to extend our framework with this relaxed setting, also yielding smaller coresets for such geometric intersection graph families, which we leave as an interesting open question.

References

1 Sara Ahmadian, Ashkan Norouzi-Fard, Ola Svensson, and Justin Ward. Better guarantees for k-means and euclidean k-median by primal-dual algorithms. In Chris Umans, editor, *58th*

IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017, pages 61–72. IEEE Computer Society, 2017.

2 Sanjeev Arora, Prabhakar Raghavan, and Satish Rao. Approximation schemes for euclidean k-medians and related problems. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, pages 106–113, New York, NY, USA, 1998. ACM. doi:10.1145/276698.276718.

3 Daniel N. Baker, Vladimir Braverman, Lingxiao Huang, Shaofeng H.-C. Jiang, Robert Krauthgamer, and Xuan Wu. Coresets for clustering in graphs of bounded treewidth. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 569–579. PMLR, 2020. URL: http://proceedings.mlr.press/v119/baker20a.html.

4 Hari Balakrishnan, Christopher L Barrett, VS Anil Kumar, Madhav V Marathe, and Shripad Thite. The distance-2 matching problem and its relationship to the mac-layer capacity of ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1069–1079, 2004.

5 Luca Becchetti, Marc Bury, Vincent Cohen-Addad, Fabrizio Grandoni, and Chris Schwiegelshohn. Oblivious dimension reduction for k-means: beyond subspaces and the johnson-lindenstrauss lemma. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1039–1050, 2019.

6 Amariah Becker, Philip N Klein, and David Saulpic. Polynomial-time approximation schemes for k-center, k-median, and capacitated vehicle routing in bounded highway dimension. In *26th Annual European Symposium on Algorithms (ESA 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

7 Ahmad Biniaz. Plane hop spanners for unit disk graphs: Simpler and better. *Comput. Geom.*, 89:101622, 2020. doi:10.1016/j.comgeo.2020.101622.

8 Vladimir Braverman, Shaofeng H-C Jiang, Robert Krauthgamer, and Xuan Wu. Coresets for clustering in excluded-minor graphs and beyond. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2679–2696. SIAM, 2021.

9 Jaroslaw Byrka, Thomas W. Pensyl, Bartosz Rybicki, Aravind Srinivasan, and Khoa Trinh. An improved approximation for k-median and positive correlation in budgeted optimization. *ACM Trans. Algorithms*, 13(2):23:1–23:31, 2017.

10 Timothy M. Chan and Dimitrios Skrepetos. Approximate shortest paths and distance oracles in weighted unit-disk graphs. *J. Comput. Geom.*, 10(2):3–20, 2019. doi:10.20382/jocg.v10i2a2.

11 Moses Charikar, Sudipto Guha, Éva Tardos, and David B. Shmoys. A constant-factor approximation algorithm for the k-median problem (extended abstract). In *Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, pages 1–10, 1999.

12 Moses Charikar and Shi Li. A dependent lp-rounding approach for the k-median problem. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part I*, volume 7391 of *Lecture Notes in Computer Science*, pages 194–205. Springer, 2012.

13 Ke Chen. On coresets for k-median and k-means clustering in metric and euclidean spaces and their applications. *SIAM Journal on Computing*, 39(3):923–947, 2009.

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

15 Vincent Cohen-Addad, Anupam Gupta, Amit Kumar, Euiwoong Lee, and Jason Li. Tight FPT approximations for k-median and k-means. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, volume 132 of *LIPIcs*, pages 42:1–42:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.ICALP.2019.42.

16 Vincent Cohen-Addad, Philip N. Klein, and Claire Mathieu. Local search yields approximation schemes for k-means and k-median in euclidean and minor-free metrics. *SIAM J. Comput.*, 48(2):644–667, 2019. doi:10.1137/17M112717X.

17 Vincent Cohen-Addad, Kasper Green Larsen, David Saulpic, and Chris Schwiegelshohn. Towards optimal lower bounds for k-median and k-means coresets. In Stefano Leonardi and Anupam Gupta, editors, *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, pages 1038–1051. ACM, 2022. doi:10.1145/3519935.3519946.

18 Vincent Cohen-Addad, David Saulpic, and Chris Schwiegelshohn. A new coreset framework for clustering. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 169–182. ACM, 2021. doi:10.1145/3406325.3451022.

19 Dan Feldman and Michael Langberg. A unified framework for approximating and clustering data. In Lance Fortnow and Salil P. Vadhan, editors, *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pages 569–578. ACM, 2011. doi:10.1145/1993636.1993712.

20 Dan Feldman and Michael Langberg. A unified framework for approximating and clustering data. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 569–578, 2011.

21 Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, pca, and projective clustering. *SIAM Journal on Computing*, 49(3):601–657, 2020.

22 Gereon Frahling and Christian Sohler. Coresets in dynamic geometric data streams. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 209–217, 2005.

23 Zachary Friggstad, Mohsen Rezapour, and Mohammad R. Salavatipour. Local search yields a PTAS for k-means in doubling metrics. *SIAM J. Comput.*, 48(2):452–480, 2019. doi:10.1137/17M1127181.

24 Jie Gao and Li Zhang. Well-separated pair decomposition for the unit-disk graph metric and its applications. *SIAM Journal on Computing*, 35(1):151–169, 2005.

25 Sariel Har-Peled and Akash Kushal. Smaller coresets for k-median and k-means clustering. *Discret. Comput. Geom.*, 37(1):3–19, 2007.

26 Sariel Har-Peled and Soham Mazumdar. On coresets for k-means and k-median clustering. In László Babai, editor, *Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004*, pages 291–300. ACM, 2004.

27 Lingxiao Huang, Shaofeng H-C Jiang, Jian Li, and Xuan Wu. Epsilon-coresets for clustering (with outliers) in doubling metrics. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 814–825. IEEE, 2018.

28 Lingxiao Huang and Nisheeth K. Vishnoi. Coresets for clustering in euclidean spaces: importance sampling is nearly optimal. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 1416–1429. ACM, 2020. doi:10.1145/3357713.3384296.

29 Kamal Jain and Vijay V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *J. ACM*, 48(2):274–296, 2001.

30 Ken-ichi Kawarabayashi, Christian Sommer, and Mikkel Thorup. More compact oracles for approximate distances in undirected planar graphs. In Sanjeev Khanna, editor, *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pages 550–563. SIAM, 2013. doi:10.1137/1.9781611973105.40.

31 Fabian Kuhn, Tim Nieberg, Thomas Moscibroda, and Rogert Wattenhofer. Local approximation schemes for ad hoc and sensor networks. In *Proceedings of the 2005 joint workshop on Foundations of mobile computing*, pages 97–103, 2005.

32 Amit Kumar, Yogish Sabharwal, and Sandeep Sen. Linear-time approximation schemes for clustering problems in any dimensions. *J. ACM*, 57(2):5:1–5:32, 2010.

33 Michael Langberg and Leonard J Schulman. Universal ε-approximators for integrals. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 598–607. SIAM, 2010.

34 Emmanuelle Lebhar and Zvi Lotker. Unit disk graph and physical interference model: Putting pieces together. In *2009 IEEE International Symposium on Parallel & Distributed Processing*, pages 1–8. IEEE, 2009.

35 Shi Li and Ola Svensson. Approximating k-median via pseudo-approximation. *SIAM J. Comput.*, 45(2):530–547, 2016.

36 Xiang-Yang Li, Gruia Calinescu, and Peng-Jun Wan. Distributed construction of a planar spanner and routing for ad hoc wireless networks. In *Proceedings. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1268–1277. IEEE, 2002.

37 Xiang-Yang Li, Wen-Zhan Song, and Yu Wang. Efficient topology control for ad-hoc wireless networks with non-uniform transmission ranges. *Wireless Networks*, 11(3):255–264, 2005.

38 Frank Schulz. Modeling sensor and ad hoc networks. In *Algorithms for Sensor and Ad Hoc Networks*, pages 21–36. Springer, 2007.

39 Amin Shahraki, Amir Taherkordi, Øystein Haugen, and Frank Eliassen. Clustering objectives in wireless sensor networks: A survey and research direction analysis. *Computer Networks*, 180:107376, 2020.

40 Christian Sohler and David P Woodruff. Strong coresets for k-median and subspace approximation: Goodbye dimension. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 802–813. IEEE, 2018.