IMPROVED APPROXIMATION ALGORITHMS FOR EUCLIDEAN FAIR k-CENTER CLUSTERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Let S be a set of points partitioned into m demographic groups. Given an upper bound k_ℓ on the number of centers that can be chosen from each group ℓ, the *fair* k-center problem asks for a set of centers C that satisfies these fairness constraints while minimizing the maximum distance from any point to its nearest center. In this paper, we first present a randomized algorithm for the Euclidean setting that achieves an approximation ratio of $1 + \sqrt{3} \approx 2.732$, thereby breaking the factor-3 barrier known for general metrics. The key idea is to reduce the problem to a generalized exact matching problem on a carefully designed auxiliary graph, which can then be solved using a generalized exact matching algorithm. For the case $m = 2$, we then develop a deterministic algorithm with the same approximation ratio and improved running time. This algorithm exploits the geometric structure of the Euclidean setting, allowing us to use maximum matching rather than exact matching. To the best of our knowledge, this is the first approximation ratio strictly below 3 for Euclidean fair k-center, suggesting that fair k-center admits better approximation guarantees in Euclidean spaces. Finally, we complement our theoretical results with experiments on five real-world datasets and one synthetic dataset with known optimal solutions, demonstrating the improvements in clustering utility over state-of-the-art methods.

1 INTRODUCTION

Clustering, as a classical machine learning algorithm, is widely used in decision-making tasks, such as unsupervised learning (Huang et al., 2019), data summarization (Gadekar et al., 2025) and feature selection (Hancer et al., 2020). It is therefore important to ensure that such algorithms are fair and do not exhibit bias toward or against any specific groups in the population (Bera et al., 2019). Concerns have been raised that bias and unfairness may propagate through automated decision-making processes in numerous human-centric domains, particularly credit, education, and social media. A typical example was provided by Kay et al. (2015), who found that Google Image search results displayed a much higher proportion of male CEOs than exist in reality. This example illustrates how biased selection in representative or influential-user identification tasks can translate into discrimination against individuals from particular racial, gender, or socioeconomic groups. In response, a growing body of research Bera et al. (2019); Huang et al. (2019); Harb & Lam (2020); Jones et al. (2020); Chiplunkar et al. (2020); Wu et al. (2024); Zhang et al. (2024); Gadekar & Thejaswi (2026); Guo et al. (2026a) has focused on developing clustering algorithms that incorporate fairness constraints for core clustering problems.

In this paper, we consider the k-center clustering problem, which has a long and rich history and has been studied under various distance metrics, including general metric spaces (Gonzalez, 1985), Euclidean spaces (Huang et al., 2024), and doubling metrics (Ceccarello et al., 2024). The classical algorithms for this problem achieve a 2-approximation (Hochbaum & Shmoys, 1985; Gonzalez, 1985). For arbitrary metric spaces, it is hard to approximate the k-center problem within a factor smaller than 2 unless $\mathcal{P} = \mathcal{NP}$ (Hsu & Nemhauser, 1979). By contrast, the Euclidean k-center problem is $\mathcal{NP}$-hard to approximate within a factor smaller than 1.822 (Bern & Eppstein, 1997), while the best-known approximation ratio remains 2. Moreover, Feder & Greene (1988) proposed a two-phase algorithm with running time $O(n \log k)$ for low-dimensional Euclidean spaces.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The fair k-center clustering has been applied to decision-making representative tasks (Kleindessner et al., 2019; Angelidakis et al., 2022), where fairness is enforced by controlling the number of selected objects from each sensitive group in the output. Given a dataset S of size n divided into m disjoint groups, denoted as $S = S_1 \cup \ldots \cup S_m$. Our goal is to select k centers to minimize the maximum distance from each point to its nearest center, while the number of centers chosen from each group S_ℓ is bounded by k_ℓ. A number of approximation algorithms have been developed for this problem, including those of Chen et al. (2016); Kleindessner et al. (2019); Jones et al. (2020). The current state-of-the-art result (Jones et al., 2020) achieves a 3-approximation in general metric spaces. Given prior works on Euclidean clustering (Nagarajan et al., 2020; Czumaj et al., 2025), it remains open whether the geometric structure of Euclidean spaces could be exploited to obtain an approximation ratio better than 3 for the fair k-center problem, thereby improving upon the best-known guarantee. To fill the gap, we propose a novel fair k-center algorithm based on Euclidean space, achieving an improved approximation ratio of $1 + \sqrt{3} \approx 2.732$.

Related work. The fair k-center problem has been extensively studied due to its wide range of applications. Several notions of fairness have been proposed, together with corresponding approximation algorithms. For example, representative fairness aims to ensure proportional representation of different groups among the selected centers (Kleindessner et al., 2019; Jones et al., 2020; Chiplunkar et al., 2020; Lin et al., 2024; Gadekar et al., 2025), which is commonly used in data summarization tasks and is therefore also referred to as data summarization fairness. Group or colorful fairness requires each cluster to contain an approximately balanced number of points from each group or color (Chierichetti et al., 2017; Bandyapadhyay et al., 2019; Backurs et al., 2019; Anegg et al., 2022; Jia et al., 2022; Ceccarello et al., 2024; Wu et al., 2024). In contrast, individual fairness focuses on ensuring that similar data points are treated similarly by the clustering outcome (Jung et al., 2020; Mahabadi & Vakilian, 2020; Negahbani & Chakrabarty, 2021; Vakilian & Yalciner, 2022).

We focus on the *group fair* k-center problem, which constrains the number of centers selected from each group. The matroid center problem with fairness, a generalization of the k-center problem in which the selected centers must satisfy a matroid constraint rather than a simple cardinality constraint, was first studied by Chen et al. (2016). They proposed a 3-approximation algorithm for general metric spaces, albeit with a high time complexity of $\Omega(n^2 \log n)$. To improve efficiency, Kleindessner et al. (2019) introduced the fair k-center clustering problem in the context of data summarization, achieving a running time of $O(nkm^2 + km^4)$ with an approximation ratio of $O(3 \cdot 2^{m-1} - 1)$. Subsequently, Jones et al. (2020) proposed an algorithm with an approximation ratio of 3 for general m groups, while reducing the running time to $O(nk + n\sqrt{k}\log k)$. Later, Angelidakis et al. (2022) introduced a unified framework combining representative k-center and private k-center formulations. More recently, Chen et al. (2024) studied the diversity-aware fair k-supplier problem, and Gadekar et al. (2025) further improved efficiency by proposing a 3-approximation algorithm for the fair k-supplier problem with a running time of $O\big((kn + k^2\sqrt{k})\log n \log k\big)$. Recently, a lot of papers have focused on the streaming model of the fair k-center problem Chiplunkar et al. (2020); Lin et al. (2024); Guo et al. (2026a;b), but the stream data setting makes their approximation ratio at least $3 + \epsilon$. Despite this progress, the approximation ratio for representative fairness in k-center clustering has remained at 3 for a long time. Building on these results, our work focuses on the Euclidean setting and aims to improve the approximation ratio for fair k-center clustering by leveraging geometric structure.

Our contributions can be summarized as follows:

- By leveraging the geometric structure of Euclidean FkC, we design a randomized algorithm that achieves a $(1 + \sqrt{3})$-approximation for any fixed number of groups m based on a reduction to the generalized exact matching problem.
- By further exploiting the geometric structure of the problem and reducing the relevant selection task to maximum matching, we develop a deterministic algorithm for $m = 2$ that achieves a ratio of $(1 + \sqrt{3})$ and a runtime of $O(kn^2)$.
- We complement our algorithms with experiments on synthetic and real-world datasets. The results show that our algorithms achieve clustering costs almost better than those of the baselines.

To the best of our knowledge, our $(1+\sqrt{3})$-approximation is the first result to break the approximation barrier of 3 for Euclidean FkC. This improves upon the best-known ratio of 3 for fair k-center in

general metrics (Chen et al., 2016; Jones et al., 2020), and demonstrates that the Euclidean setting admits strictly better approximability. This separation is particularly notable in light of the classical k-center problem. For standard, unconstrained k-center, the best possible approximation ratio is 2 in general metrics, and the same ratio remains the best known even in Euclidean space, including the two-dimensional plane. Thus, while Euclidean geometry does not currently improve the approximation ratio for standard k-center, our results show that it can be exploited to improve the approximation guarantee for the fair variant.

Moreover, the deterministic algorithm is significant because it avoids relying on generalized exact matching as a black box. While exact matching and its generalizations are powerful tools, designing efficient deterministic polynomial-time algorithms for them remains a long-standing challenge. By reducing the key subproblem to maximum matching, we obtain a deterministic algorithm with the same $(1+\sqrt{3})$-approximation guarantee and a simple polynomial running time.

2 PRELIMINARIES

Let S be a finite set of n data points in Euclidean space, where $d : S \times S \to \mathbb{R}_{\geqslant 0}$ is the distance on S. We denote $d(s, C) = \min_{c \in C} d(s, c)$ as the distance between a point s and the center set C, and then $d(S, C) = \max_{s \in S} d(s, C)$. If $d(S, C) \leqslant \lambda$, we say that the set C *covers* S within radius λ. For convenience, we use $B(s, \lambda) = \{x \mid x \in S, d(s, x) \leqslant \lambda\}$ to denote the set of points of $x \in S$ that are within distance λ from s. Then, $B(\Gamma, \lambda) = \bigcup_{s \in \Gamma} B(s, \lambda)$.

Given a parameter $k \in \mathbb{N}$, the traditional (standard) k-center clustering problem is to select a set of k points $C \subseteq S$ to serve all points in S such that $d(S, C)$ is minimized. Following previous work Guo et al. (2026b), we formally define the fair k-center problem as follows. The fair k-center clustering problem is a generalization of the traditional k-center problem, which requires satisfying further fairness constraints. Let the data set S be composed of m *disjoint* groups. For given fair constraint k_ℓ for each group ℓ and the cardinality constraint $k = \sum_{\ell=1}^{m} k_\ell$, the *fair k-center* problem (FkC) is to find a center set C as follows:

$$\begin{aligned} \min_{C \subseteq S} \quad & d(S, C) \\ s.t. \quad & |\, C_\ell \,| \leqslant k_\ell, \ \forall \ell \\ & |\, C \cap S \,| = \sum_{\ell=1}^{m} |\, C \cap S_\ell \,| \leqslant k, \end{aligned}$$

where $C_\ell = C \cap S_\ell$. Besides, we use analogous notation throughout the paper, e.g., $\Gamma_\ell = \Gamma \cap S_\ell$.

Following common practice in Guha (2009); Guo et al. (2025), we assume that the optimal radius r^* is known throughout the paper for standard and streaming k-center clustering. Moreover, for simplicity, we assume that $r^* = 1$ without loss of generality.

As the first component of our algorithm, we adopt the λ-independent center set, as defined below, which has been extensively studied in the literature on streaming fair k-center clustering Chiplunkar et al. (2020); Guo et al. (2026b):

Definition 1. *(λ-independent center set) Γ is a (maximal) λ-independent center set of S iff*

(1) The distance between any two points $p, q \in \Gamma$ is larger than λ, i.e. $d(p, q) > \lambda$.

(2) For any $p \in S$, there exists a point $q \in \Gamma$ such that $d(p, q) \leqslant \lambda$.

When setting $\lambda = 2$, it can be shown that the λ-independent center set Γ satisfies $|\Gamma| \leqslant k$; that is, Γ contains at most k points. Therefore, Γ yields a 2-approximation for the standard k-center problem. Moreover, using this component as a building block, we can construct an algorithm with an approximation ratio of $\sqrt{3}$ in Euclidean space by setting $\lambda = \sqrt{3}$, which guarantees that the resulting independent center set contains at most $2k$ points. For brevity, we use $\sigma(S, \lambda)$ to denote the λ-independent center set of S. However, under fairness constraints, such an independent center set may not constitute a feasible solution.

We transform our problem into the outsourcing fair k-center (OF-k-SUP) problem:

Definition 2. *Given a set of data points Λ to be covered and a collections of disjoint sets of outsourcing points P_j, $j \in [m]$, in Euclidean space. For positive integer τ, and k_j, j and k_2, the goal is to find a center set $C \subseteq \Lambda \cup P$, $P \bigcup_j P_j$ such that:*

- *The objective $d(\Lambda, C)$ is minimized, i.e., the maximal distance from any point $u \in \Lambda$ to its nearest center is minimized.*
- *Both $|C \cap \Lambda| \leqslant \tau$ and $|C \cap P| \leqslant k_j$ for $j \in [m]$ hold.*

Note that the OF-k-SUP problem generalizes both k-supplier and k-center: the problem reduces to k-supplier when $\tau = 0$ and $m = 1$, and to k-center when $k_j = 0$ for $\forall j$.

3 A RANDOMIZED ALGORITHM FOR FkC

In this section, we will develop a randomized algorithm for fair k-center (FkC) by employing the colored budgeted matching. To this end, we first present an algorithm to reduce FkC to a special version of OF-k-SUP, together with the geometric property that motivates the core idea of our algorithm. Then, we give the method that further transforms the special version of OF-k-SUP to the colored budgeted perfect matching problem, which is a generalization of the well-known exact matching problem. Lastly, we provide the randomized algorithms for colored budgeted matching in Appendix C.

3.1 REDUCING A SPECIAL VERSION OF OF-k-SUP AND GEOMETRIC PROPERTY

The key idea for reducing FkC to a special case of OF-k-SUP is to first compute a $\sqrt{3}$-independent center set $\Gamma = \sigma(S, \sqrt{3})$ without considering the fairness constraints. Instead of directly covering the entire dataset S, we then seek centers from the groups S_ℓ to cover the representative set Γ while respecting the group-wise budgets.

Specifically, set $\Lambda = \Gamma$ and $P_j = S_j$ for each $j \in [m]$. By setting $\tau = 0$, this immediately yields an instance of OF-k-SUP with group budgets k_j.

Importantly, the resulting OF-k-SUP instance has additional structure, captured by the following geometric property.

Lemma 1 (Nagarajan et al. (2020))**.** *Let $\Gamma = \sigma(S, \sqrt{3})$. Then, for any $s \in S$, the unit ball $B(s, 1)$ contains at most two points of Γ.*

By Lem. 1, any unit-radius cluster centered at a point of S can contain at most two representatives of Γ. Hence, the coverage induced by a candidate center can be represented by either a single representative or a pair of representatives, with the latter naturally encoded by an edge. Equivalently, in the resulting OF-k-SUP instance, each candidate point in P_j (i.e., S_j) needs to cover at most two points of Γ.

The remaining task is therefore to select a set of actual centers $C \subseteq S$ such that each selected center covers at most two representatives and, collectively, the centers cover all points of Γ within radius 1, while satisfying the group-wise budget constraints $|C \cap S_\ell| \leqslant k_\ell$ for every $\ell \in [m]$. Since $\sum_{\ell=1}^{m} k_\ell = k$, these group-wise constraints immediately imply $|C| \leqslant k$. We therefore omit the global cardinality constraint $|C| \leqslant k$ throughout this section for brevity.

3.2 FURTHER TRANSFORMATION TO COLORED BUDGETED PERFECT MATCHING

We next show how to reduce the selection of a desirable center set C to a colored budgeted perfect matching problem. To this end, we construct an auxiliary colored graph $G = (V, E)$ as follows.

For the vertex set, we create one vertex v for each representative $v \in \Gamma$, together with a dummy copy v'. Thus, $V = \Gamma \cup \Gamma'$, where $\Gamma' = \{v' : v \in \Gamma\}$.

We then introduce three types of edges. First, for any two representatives $u, v \in \Gamma$, we add a color-ℓ edge between u and v if there exists a point $x \in S_\ell$ such that $d(x, u) \leqslant 1$ and $d(x, v) \leqslant 1$. Such an edge represents selecting one center from group ℓ that simultaneously covers both representatives.

Second, for each representative $v \in \Gamma$, we add a color-ℓ edge between v and its dummy copy v' if there exists a point $x \in S_\ell$ with $d(x, v) \leqslant 1$. This edge represents selecting one center from group ℓ to cover only v. Third, we add an uncolored edge between every pair of dummy vertices $u', v' \in \Gamma'$. We denote these three edge sets by E^2, E^1, and E^0, respectively. The dummy vertices and the edges in E^0 serve only as completion gadgets: once all representatives in Γ are saturated by edges in $E^1 \cup E^2$, the unmatched dummy vertices can be paired through edges in E^0 to complete the matching into a perfect matching, without selecting additional centers or consuming any group budget.

We assign weight j to every edge in E^j, for $j \in \{0, 1, 2\}$. Hence, the edge weight exactly records the number of representatives of Γ covered by the corresponding edge. In particular, because a matching cannot cover any representative more than once, its total weight is at most $|\Gamma|$, and a matching of weight exactly $|\Gamma|$ saturates all representatives. Meanwhile, each colored edge corresponds to the selection of one actual center from the associated group, so imposing at most k_ℓ edges of color ℓ enforces the group-wise budget $|C \cap S_\ell| \leqslant k_\ell$.

Therefore, the desired center-selection problem is equivalent to finding a perfect matching of total weight $|\Gamma|$ that uses at most k_ℓ edges of color ℓ for every $\ell \in [m]$. The detailed construction of the auxiliary graph is given in Alg. 1 in Appendix B.1.

The following lemma establishes the connection between the desirable center set C for covering Γ and the perfect color-budgeted matching problem in the auxiliary graph G.

Lemma 2. *The representative set Γ can be covered by a center set $C \subseteq S$ within optimum radius 1, where $|C \cap S_\ell| \leqslant k_\ell$ holds for each $\ell \in [m]$ iff the corresponding auxiliary graph $G = (V, E)$ contains a matching M with weight $|\Gamma|$, such that M uses at most k_ℓ edges with color ℓ.*

Following the above lemma, we need only to compute such a perfect matching M that is with weight $w(M) = |\Gamma|$ and contains at most k_ℓ edges with color ℓ. Note that even when $m = 2$, finding such a matching contains the exact matching problem as a special case when $k_1 + k_2 = k$. In the exact matching problem, we are given a graph $G = (V, E)$, where each edge is either blue or red. The task is to find a perfect matching with exactly k_r red edges for a given integer k_r. It is well known that the exact matching problem admits a randomized polynomial algorithm Mulmuley et al. (1987), while developing a deterministic polynomial algorithm for this problem remains open El Maalouly et al. (2024). To complete the result, we provide the algorithms and the correctness proof for colored budgeted matching in Appendix C. We note that our algorithm for colored budgeted perfect matching is an extension of the classical algorithm for exact matching due to Mulmuley et al. (1987).

4 The Deterministic Algorithm with Approximation Ratio $1 + \sqrt{3}$

In this section, we present a deterministic $(1 + \sqrt{3}) \approx 2.732$ approximation algorithm for the case $m = 2$. The key idea is to reduce FkC to a more structured instance of OF-k-Sup. Starting from $C_0 = \varnothing$, we iteratively add centers until all points in $S \backslash B(C_0, 2)$ belong to a single group, while preserving the required approximation guarantee. We then formulate the resulting OF-k-Sup instance as a maximum matching problem, from which the desired $(1 + \sqrt{3})$ approximation follows.

4.1 Transformation to OF-k-Sup

For transforming to OF-k-Sup, we grow C_0 based on the following lemma.

Lemma 3. *Let $S = S_i \dot{\cup} S_j$ with $i \neq j$, and let $k = k_i + k_j$, where $S = S_i \dot{\cup} S_j$ means $S = S_i \cup S_j$ while $S_i \cap S_j = \varnothing$. Assume that S can be partitioned into k unit-radius balls, where at most k_i points from S_i and at most k_j points from S_j can cover these k balls with radius 2. Let $\Gamma = \sigma(S, 2)$. Suppose that $|\Gamma \cap S_i| < k_i$ and $|\Gamma \cap S_j| > k_j$. Then, for any $u \in \Gamma \cap S_i$, we have that $S \backslash B(u, 2)$ can be covered by:*

- *at most $k - 1$ unit-radius balls.*
- *at most $k_i - 1$ centers from S_i and at most k_j centers from S_j, where the removed points may still be served as centers.*

Following Lem. 3, the key idea of our algorithm is to iteratively grow a center set C_0 by computing Γ, and adding Γ_ℓ with $|\Gamma_\ell| < k_\ell$ to C_0. The growth iteration terminates when $S_\ell \backslash B(C_0, 2) = \varnothing$ holds for some ℓ. The detailed algorithm is as in Alg. 2 in Appendix B.2.

Lemma 4. *Assume $S \cap S_\ell \neq \varnothing$ for each $\ell \in \{1, 2\}$. If S contains no Γ with $|S_\ell \cap \Gamma| \geqslant 1$ holds for $\forall \ell$, then any point of S_ℓ covers all points of $S_{3-\ell}$ for each $\ell \in \{1, 2\}$ with radius 2.*

Proof. Suppose there exists some point $u \in S_\ell$ that does not cover a point v in $S_{3-\ell}$. Then, we can simply construct $\Gamma \supseteq \{u, v\}$, which satisfies $|S_\ell \cap \Gamma| \geqslant 1$. □

Therefore, when there exists no Γ with $|S_\ell \cap \Gamma| \geqslant 1$, we need only to randomly select two points u_1 and u_2, where u_ℓ is from S_ℓ, because such u_1 and u_2 respectively cover all remaining points of S_1 and S_2 by Lem. 4. Putting things together, we have the following lemma for the correctness of Alg. 2.

Theorem 1. *Alg. 2 runs in time $O(kn^2)$, and produces Λ and C_0 satisfying the claims below:*

(1) The points of Λ appear in at most $k - |C_0|$ clusters of the optimal solution $\mathcal{O}$, which can be covered by $k - |C_0|$ centers (with radius 2), of which $k_\ell - |C_0 \cap S_\ell|$ centers are from S_ℓ.

(2) There exists $\ell \in \{1, 2\}$, such that $\Lambda \cap S_\ell = \varnothing$.

For brevity, we assume Λ produced by Alg. 2 contains only points from S_1. Then the task remains to use at most k_2 points of S_2 and k_1 points of Λ to cover all the points of $\Lambda = S \backslash B(C_0, 2) \subseteq S_1$.

4.2 Covering the remaining points Λ

Our algorithm for covering Λ is based on the property of Λ as follows:

Lemma 5. *Let $\Gamma = \sigma(\Lambda, \sqrt{3})$. For any point $o \in S_2$, $B(o, 1)$ contains at most one point of Γ.*

Then, a natural idea is to use points of S_1 to cover most points of Λ by maximum matching, and then use points of S_2 to cover the remaining points of Λ. However, there remains a problem: Λ may contain a subset of points, say $\Lambda_0 \subseteq \Lambda$, that cannot be covered by any point of S_2 within a unit radius. Then, if the remaining points contain points of Λ_0, the algorithm is not feasible. So the key of our algorithm is to first cover Λ_0, and then the other part of Λ.

For covering Λ_0, we need only to simply use $\sigma(\Lambda_0, 2)$ following the lemma below:

Lemma 6. *Assume that Λ admits a cover by k unit-radius balls whose centers belong to $\Lambda \cup S_2$, with at most k_1 chosen from Λ and at most k_2 chosen from S_2. Let $\Lambda_0 = \{u \mid u \in \Lambda,\ d(u, v) > 1 \text{ for } \forall v \in S_2\}$ and $\Gamma_0 = \sigma(\Lambda_0, 2)$. Then $\Lambda \backslash B(\Gamma_0, 2)$ admits a cover by at most $k - |\Gamma_0|$ unit-radius balls whose centers are in $\Lambda \cup S_2$, using at most $k_1 - |\Gamma_0|$ centers from Λ and at most k_2 centers from S_2.*

For covering the remaining points of $\Lambda := \Lambda \backslash B(\Gamma_0, 2)$, we first compute $\Gamma = \sigma(\Lambda, \sqrt{3})$. Then, we use k_1 points of Λ to cover most points of Γ and lastly cover the remaining points of Λ by S_2. Following Lem. 1, each point of S, and hence every point of Λ, can cover at most two points of Γ within unit radius. Similar to the task for fixed m, we construct an auxiliary graph $G(V, E)$ by setting $V = \Gamma$ and construct E as below:

> **for** *every pair of points $u, v \in \Gamma$* **do**
> Add edge $e(u, v)$ to E if u and v can be covered by a point $o \in \Lambda$ within a radius $r^* = 1$, i.e., $d(u, o) \leqslant 1$ and $d(v, o) \leqslant 1$ both hold.

For using maximum matching M to produce the center set according to Γ, we have

Then, by the preceding lemma, we first compute a maximum matching M in G. For the matched edges, we use $\min\{k_1, |M|\}$ centers from S_1 to cover the corresponding pairs of points in Γ, with each center covering two points. The remaining points of Γ are then covered individually. Specifically, if $k_1 > |M|$, we use up to $k_1 - |M|$ additional centers from S_1, namely points in Λ. We then use at most k_2 centers from S_2 to cover the remaining points; this is valid because every point of the current Λ, after removing Λ_0, has a neighbor in S_2 within unit radius. By the preceding lemma, these centers are sufficient to cover all remaining points of Γ within the optimal unit radius. The detailed algorithm is presented in Alg. 3 in Appendix B.3.

Lemma 7. *Let M be a maximum matching in the auxiliary graph $G = (\Gamma, E)$, and $V(M)$ be the set of points in M. Then:*

- *If $|M| \leqslant k_1$, then $|\Gamma \backslash V(M)| \leqslant k_1 - |M| + k_2$.*
- *If $|M| > k_1$, then for any $M' \subseteq M$ with $|M'| = k_1$, we have $|\Gamma \backslash V(M')| \leqslant k_2$.*

Lemma 8. *Alg. 3 runs in time $O(n^2 + k^{5/2})$ and produces C for the remaining instance of Euclidean fair k center regarding Λ, C_0, k, such that: (1) $|C| \leqslant k$ and $|C \cap S_\ell| \leqslant k_\ell$; (2) for every $u \in \Lambda$, $d(C, u) \leqslant 1 + \sqrt{3}$.*

Combining Thm. 1 and Lem. 8, we eventually have the following theorem:

Theorem 2. *The Euclidean fair k-center problem admits a deterministic algorithm with approximation ratio $1 + \sqrt{3}$ and runtime $O(kn^2)$.*

5 Evaluation

In this section, we evaluate the performance of our methods on a variety of real-world and synthetic datasets and report the clustering cost and runtime to compare against the baseline methods.

Datasets. We evaluate all algorithms on six datasets, including five real-world datasets and one synthetic dataset for which the optimal solution is known and the generating method is provided by Kleindessner et al. (2019). Following prior work Jones et al. (2020); Chen et al. (2019), we use the five real-world datasets Wholesale, Student, Bank, CreditCard and Adult. The dataset details are described below, and summary statistics are reported in Table 2. For each dataset, we consider each listed categorical attribute separately as the group attribute, thereby generating different fair k-center instances. The values in the table report the number of points belonging to each group.

Algorithms. First, we have the traditional k-center clustering algorithm (*Gonzalez (1985)*) as a baseline for comparison, which is a 2-approximation algorithm and its result does not satisfy the fairness constraints. Following closely related work, existing approximation algorithms for fair k-center clustering mainly include:

- *Kleindessner et al. (2019)*: they proposed a 5-approximation algorithm for fair k-center clustering in metric spaces when $m = 2$, and a $(3 \cdot 2^{m-1} - 1)$-approximation algorithm for general m;
- *Jones et al. (2020)*: they proposed a 3-approximation algorithm for fair k-center clustering in metric spaces;
- *Lin et al. (2026)*: they proposed a $(1 + \sqrt{3})$-approximation algorithm for fair k-center clustering in Euclidean spaces by fixed parameter k and m.

In addition, we evaluate the two algorithms proposed in this paper: RdEu (in Section 3), a randomized $(1 + \sqrt{3})$-approximation algorithm for fair k-center clustering with a fixed number of groups m in Euclidean space, and DeEu (in Section 4), a deterministic $(1 + \sqrt{3})$-approximation algorithm for the case $m = 2$.

Experiment setup. We evaluate the algorithms on both real-world and synthetic datasets with varying dataset sizes and different values of the parameters k and m, and compare their average clustering costs. For synthetic datasets, we report the empirical approximation ratio achieved by each algorithm to validate the effectiveness of our proposed theoretical method. For the *real-world datasets*, we report the clustering cost under different experimental settings, with all datasets normalized using MinMax scaling.

Following the previous impact fairness principle of Feldman et al. (2015), we restrict the selection to k_ℓ data points from the ℓ-th group as centers, which sets the required number of fair center selections from each group proportional to the group size. In our experiments, we use the Euclidean distance as the default metric, as it is commonly adopted in this setting. All experiments were conducted on an Apple M1 Max CPU with 32 GB of RAM using Python 3.8. Each reported value is averaged over at least 10 runs for each setting. We evaluate *cost* in terms of k-center clustering cost for real-world datasets and the *Empirical Approximation Ratio* for synthetic datasets. In addition, we report the running time in *seconds*.

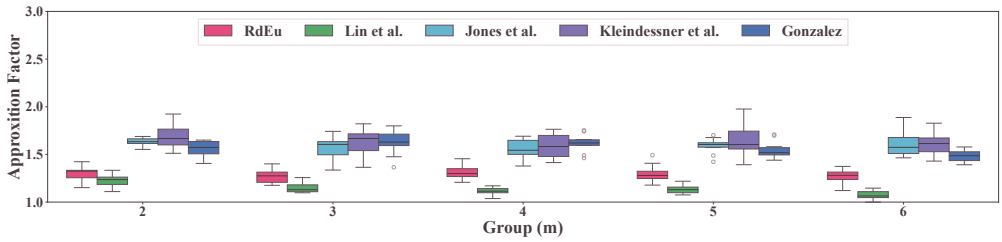

Figure 1: Approximation Factor (cost/r^*) of our algorithms (RDEU & DEEU) compared with baselines. Varying number of groups m with dataset size $|S| = 200$.

5.1 EXPERIMENT RESULTS

In this section, we first report experimental results on the synthetic dataset with an optimal solution fixed size of $k = 25$, under varying dataset sizes $|S| = \{2K, 4K, 6K, 8K, 10K\}$, while fixing $m = 2$ and varying the number of groups m from 2 to 6 while fixing $|S| = 200$ and an optimal solution fixed size of $k = 9$. We then report the performance of our algorithm on 5 real-world datasets with different fair features to evaluate the comparison of the different algorithms with fixed and large center sizes k and the unbalanced fairness's results. Additional experimental results on unbalanced group fairness constraints, larger values of k, and running time are presented in the appendix.

5.1.1 CLUSTERING QUALITY ON SYNTHETIC DATA

The synthetic dataset was constructed so that the optimal solution to the fair k-center problem is known. We define the maximum observed approximation factor as the empirical approximation ratio. As shown in Fig. 1, *all experimental results on the synthetic dataset are consistent with the theoretical guarantee*: across different numbers of groups m, the *empirical approximation ratio* of our algorithms, RDEU and DEEU, is clearly bounded by 2.732 to match our theoretical analysis. We next highlight the additional observations supported by our results.

As shown in Fig. 1, our algorithms also achieve lower average empirical approximation ratios than the unconstrained 2-approximation algorithm of Gonzalez, despite the latter having a stronger worst-case approximation guarantee. This does not contradict the theoretical bounds, which characterize worst-case performance rather than the relative behavior of the algorithms on specific instances. A possible explanation is that our algorithms first construct a carefully selected candidate set containing up to $2k$ representative points and then exploit the geometric structure and fairness constraints when selecting the final centers. During this processing, we collect a richer candidate set, which provides additional flexibility in center selection and can lead to solutions of substantially lower cost on typical instances, even though the algorithm must additionally satisfy the fairness constraints.

In addition, although *Kleindessner et al.* has a higher ratio, its average performance and lower bound are similar to those of Gonzalez and Jones. In contrast, our algorithm explores feasible regions to get lower cost. We believe this is because our algorithms employ the geometric structure of Euclidean space, which provides more candidate solutions and leads to better clustering performance.

For the datasets with a varying number of groups m, RDEU achieves consistently better performance than the other methods in terms of both the average and the maximum empirical approximation ratios, as shown in Fig. 1, with the exception of Lin et al.. This observation is consistent with our previous analysis, since RDEU has a better theoretical approximation guarantee than the other baseline methods. Although RDEU and the method of Lin et al. share the same approximation ratio of $1 + \sqrt{3}$, the latter achieves slightly better empirical solution quality. We attribute this difference to its use of an exhaustive search procedure for selecting the final center set, which explores the candidate space more thoroughly but incurs a substantially higher computational cost.

5.1.2 CLUSTERING QUALITY ON REAL-WORLD DATA

We evaluate all algorithms on five real-world datasets with different fairness attributes, as summarized in Table 1, where these attributes are used to define the fair groups of data points. When measuring the clustering cost, the center-set size k is set to $p\%$ of the dataset size, with $p = 1$. We report the

Algorithms	Setting	Approx. Ratio	Wholesale		Student		
			Region	Channel	Sex	School	Address
Gonzalez (1985)	Unfair	2	$0.901_{\pm 0.030}$	$0.901_{\pm 0.030}$	$1.686_{\pm 0.036}$	$1.686_{\pm 0.036}$	$1.686_{\pm 0.036}$
Lin et al. (2026)	Euclidean, FPT	$1+\sqrt{3}$	$0.656_{\pm 0.037}$	$0.684_{\pm 0.037}$	1.388	1.427	1.404
Kleindessner et al. (2019)	–	$3 \cdot 2^{m-1} - 1$	$0.872_{\pm 0.079}$	$0.884_{\pm 0.069}$	$1.785_{\pm 0.073}$	$1.791_{\pm 0.078}$	$1.744_{\pm 0.063}$
Jones et al. (2020)	–	3	$0.906_{\pm 0.016}$	$0.901_{\pm 0.030}$	$1.712_{\pm 0.072}$	$1.725_{\pm 0.090}$	$1.691_{\pm 0.054}$
DeEu	Euclidean, $m = 2$	$1+\sqrt{3}$	$0.871_{\pm 0.155}$	$0.820_{\pm 0.162}$	$1.698_{\pm 0.056}$	$1.677_{\pm 0.049}$	$1.684_{\pm 0.040}$
RdEu	Euclidean	$1+\sqrt{3}$	$0.722_{\pm 0.084}$	$0.700_{\pm 0.059}$	$1.603_{\pm 0.028}$	$1.587_{\pm 0.025}$	$1.588_{\pm 0.033}$

Algorithms	Bank	Adult		Creditcard			Avg.
	Marital	Sex	Race	Marriage	Education	Sex	
Gonzalez (1985)	$0.386_{\pm 0.006}$	$0.209_{\pm 0.001}$	$0.208_{\pm 0.001}$	$0.530_{\pm 0.003}$	$0.541_{\pm 0.003}$	$0.521_{\pm 0.004}$	$0.841_{\pm 0.560}$
Lin et al. (2026)	–	–	–	–	–	–	–
Kleindessner et al. (2019)	$0.409_{\pm 0.027}$	$0.238_{\pm 0.015}$	$0.231_{\pm 0.008}$	$0.584_{\pm 0.021}$	$0.585_{\pm 0.030}$	$0.566_{\pm 0.029}$	$0.881_{\pm 0.582}$
Jones et al. (2020)	$0.390_{\pm 0.012}$	$0.209_{\pm 0.001}$	$0.214_{\pm 0.005}$	$0.585_{\pm 0.036}$	$0.609_{\pm 0.041}$	$0.545_{\pm 0.014}$	$0.863_{\pm 0.561}$
DeEu	$0.378_{\pm 0.008}$	$0.207_{\pm 0.001}$	–	–	–	$0.520_{\pm 0.002}$	-
RdEu	$0.380_{\pm 0.000}$	$0.218_{\pm 0.001}$	$0.209_{\pm 0.000}$	$0.528_{\pm 0.001}$	$0.555_{\pm 0.002}$	$0.526_{\pm 0.001}$	$0.783_{\pm 0.520}$

Table 1: Comparison of clustering cost on real-world datasets. (Green indicates best results and lighter green indicates second-best among fair algorithms; unfair baselines are excluded).

average clustering cost over 10 runs, together with the corresponding standard deviation. The first block of Table 1 reports results on the smaller datasets and includes *Lin et al.*. In addition, DeEu is applied only to instances with a small number of groups ($m \leqslant 3$); therefore, we do not evaluate it on instances with larger numbers of groups, in accordance with its algorithmic setting. The key observations from the synthetic experiments are also reflected in the real-world datasets. Overall, in terms of average clustering cost, the empirical results are consistent with our theoretical analysis and generally demonstrate the advantages of our algorithms over the baselines.

For the smaller datasets, RdEu consistently achieves the second-best clustering cost, surpassed only by *Lin et al.*. In the second block of Table 1, whenever DeEu is applicable, it consistently achieves the best clustering cost among the fair algorithms. Moreover, its results are often close to those of the unconstrained Gonzalez algorithm, suggesting that DeEu incurs only a limited additional cost for satisfying the fairness constraints on these instances. This behavior is consistent with its algorithmic design, including the step get result from a similar method of traditional k-center method, which explicitly exploits the Euclidean structure to preserve high-quality candidate centers while enforcing the fairness requirements.

As the dataset size increases, satisfying the fairness constraints may impose a stronger restriction on the feasible center selection, which can lead to a larger clustering cost compared with the unconstrained k-center solution. This explains why the methods of Jones et al. and Kleindessner et al. generally exhibit higher costs than Gonzalez on the larger datasets. In contrast, RdEu consistently achieves either the best or the second-best result among the fair algorithms. An explanation is that RdEu first constructs a representative candidate set containing up to $2k$ points, which provides greater flexibility in selecting the final fair center set. Consequently, it can better accommodate the fairness constraints while retaining candidates with favorable geometric coverage, thereby reducing the additional clustering cost introduced by fairness.

6 Conclusion

In this paper, we present approximation algorithms for Euclidean FkC that achieve an approximation ratio of $1+\sqrt{3}$, improving the previous best ratio of 3 for general metrics. By exploiting geometric properties of an optimal solution to FkC in Euclidean space, we first design a randomized algorithm with ratio $1+\sqrt{3} \approx 2.732$ for any fixed m, via a reduction to a special instance of OF-k-Sup and then to colored budgeted matching, a generalization of exact matching. Then, observing the high running time and randomized nature of this algorithm, we propose a deterministic algorithm for the case $m = 2$ that achieves the same approximation ratio with a lower runtime $O(kn^2)$. Finally, experiments on both synthetic and real-world datasets show that our algorithm achieves quality comparable to or lower than that of state-of-the-art baselines.

AI USE STATEMENT

We used an AI language model to assist with language refinement, proofreading, and literature searches. The author(s) take full responsibility for the final manuscript and all associated theoretical claims, proofs, code, and reported experimental results.

REFERENCES

Georg Anegg, Laura Vargas Koch, and Rico Zenklusen. Techniques for generalized colorful k-center problems. In *30th Annual European Symposium on Algorithms (ESA 2022)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.

Haris Angelidakis, Adam Kurpisz, Leon Sering, and Rico Zenklusen. Fair and fast k-center clustering for data summarization. In *International Conference on Machine Learning*, pp. 669–702. PMLR, 2022.

Arturs Backurs, Piotr Indyk, Krzysztof Onak, Baruch Schieber, Ali Vakilian, and Tal Wagner. Scalable fair clustering. In *International Conference on Machine Learning*, pp. 405–413. PMLR, 2019.

Sayan Bandyapadhyay, Tanmay Inamdar, Shreyas Pai, and Kasturi Varadarajan. A constant approximation for colorful k-center. *Leibniz International Proceedings in Informatics, LIPIcs*, 144, 2019.

Suman Bera, Deeparnab Chakrabarty, Nicolas Flores, and Maryam Negahbani. Fair algorithms for clustering. *Advances in Neural Information Processing Systems*, 32, 2019.

Marshall Bern and David Eppstein. Approximation algorithms for geometric problems. *Approximation algorithms for NP-hard problems*, pp. 296–345, 1997.

Matteo Ceccarello, Andrea Pietracaprina, and Geppino Pucci. Fast and accurate fair k-center clustering in doubling metrics. In *Proceedings of the ACM Web Conference 2024*, pp. 756–767, 2024.

Danny Z Chen, Jian Li, Hongyu Liang, and Haitao Wang. Matroid and knapsack center problems. *Algorithmica*, 75:27–52, 2016.

Xianrun Chen, Sai Ji, Chenchen Wu, Yicheng Xu, and Yang Yang. An approximation algorithm for diversity-aware fair k-supplier problem. *Theoretical Computer Science*, 983:114305, 2024.

Xingyu Chen, Brandon Fain, Liang Lyu, and Kamesh Munagala. Proportionally fair clustering. In *International Conference on Machine Learning*, pp. 1032–1041. PMLR, 2019.

Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 5036–5044, 2017.

Ashish Chiplunkar, Sagar Kale, and Sivaramakrishnan Natarajan Ramamoorthy. How to solve fair k-center in massive data models. In *Proceedings of the 37th ICML*, pp. 1877–1886, 2020.

Artur Czumaj, Guichen Gao, Mohsen Ghaffari, and Shaofeng H-C Jiang. Fully scalable mpc algorithms for euclidean k-center. In *Proceedings of the 52nd International Colloquium on Automata, Languages and Programming (ICALP)*, pp. 127. Leibniz International Proceedings in Informatics (LIPICS), Schloss Dagstuhl . . . , 2025.

Nicolas El Maalouly, Sebastian Haslebacher, and Lasse Wulf. On the exact matching problem in dense graphs. volume 289, pp. 33–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024.

Tomás Feder and Daniel Greene. Optimal algorithms for approximate clustering. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, pp. 434–444, 1988.

Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 259–268, 2015.

Ameet Gadekar and Suhas Thejaswi. Capacitated fair-range clustering: Hardness and approximation algorithms. In *Forty-third International Conference on Machine Learning*, 2026.

Ameet Gadekar, Aristides Gionis, and Suhas Thejaswi. Fair clustering for data summarization: Improved approximation algorithms and complexity insights. In *Proceedings of the ACM on Web Conference 2025*, pp. 4458–4469, 2025.

Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical computer science*, 38:293–306, 1985.

Sudipto Guha. Tight results for clustering and summarizing data streams. In *Proceedings of the 12th International Conference on Database Theory*, pp. 268–275, 2009.

Longkun Guo, Chaoqi Jia, Kewen Liao, Zhigang Lu, and Minhui Xue. Near-optimal algorithms for instance-level constrained k-center clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

Longkun Guo, Chaoqi Jia, and Chao Chen. Fair k-center clustering on massive social network data streams. In *Proceedings of the ACM Web Conference 2026*, pp. 4689–4700, 2026a.

Longkun Guo, Zeyu Lin, Chaoqi Jia, and Chao Chen. Improved streaming algorithm for fair k-center clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 36946–36954, 2026b.

Emrah Hancer, Bing Xue, and Mengjie Zhang. A survey on feature selection approaches for clustering. *Artificial intelligence review*, 53(6):4519–4545, 2020.

Elfarouk Harb and Ho Shan Lam. Kfc: A scalable approximation algorithm for k-center fair clustering. *Advances in neural information processing systems*, 33:14509–14519, 2020.

Dorit S Hochbaum and David B Shmoys. A best possible heuristic for the k-center problem. *Mathematics of operations research*, 10(2):180–184, 1985.

Wen-Lian Hsu and George L Nemhauser. Easy and hard bottleneck location problems. *Discrete Applied Mathematics*, 1(3):209–215, 1979.

Lingxiao Huang, Shaofeng Jiang, and Nisheeth Vishnoi. Coresets for clustering with fairness constraints. *Advances in neural information processing systems*, 32, 2019.

Lingxiao Huang, Jian Li, and Xuan Wu. On optimal coreset construction for euclidean (k, z)-clustering. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pp. 1594–1604, 2024.

Xinrui Jia, Kshiteej Sheth, and Ola Svensson. Fair colorful k-center clustering. *Mathematical Programming*, 192(1-2):339–360, 2022.

Matthew Jones, Huy Nguyen, and Thy Nguyen. Fair k-centers via maximum matching. In *International Conference on Machine Learning*, pp. 4940–4949. PMLR, 2020.

Christopher Jung, Sampath Kannan, and Neil Lutz. Service in your neighborhood: Fairness in center location. *Foundations of Responsible Computing (FORC)*, 2020.

Matthew Kay, Cynthia Matuszek, and Sean A Munson. Unequal representation and gender stereotypes in image search results for occupations. In *Proceedings of the 33rd annual acm conference on human factors in computing systems*, pp. 3819–3828, 2015.

Matthäus Kleindessner, Pranjal Awasthi, and Jamie Morgenstern. Fair k-center clustering for data summarization. In *International Conference on Machine Learning*, pp. 3448–3457. PMLR, 2019.

Zeyu Lin, Longkun Guo, and Chaoqi Jia. Streaming fair k-center clustering over massive dataset with performance guarantee. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 105–117. Springer, 2024.

Zeyu Lin, Chaoqi Jia, Longkun Guo, and Chao Chen. Parameterized and streaming algorithms for euclidean fair k-center clustering. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence, IJCAI-26*, pp. 4545–4553, 8 2026.

Sepideh Mahabadi and Ali Vakilian. Individual fairness for k-clustering. In *International conference on machine learning*, pp. 6586–6596. PMLR, 2020.

Silvio Micali and Vijay V Vazirani. An $o(v|v|c|e|)$ algoithm for finding maximum matching in general graphs. In *21st Annual symposium on foundations of computer science (Sfcs 1980)*, pp. 17–27. IEEE, 1980.

Ketan Mulmuley, Umesh V Vazirani, and Vijay V Vazirani. Matching is as easy as matrix inversion. In *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pp. 345–354, 1987.

Viswanath Nagarajan, Baruch Schieber, and Hadas Shachnai. The euclidean k-supplier problem. *Mathematics of Operations Research*, 45(1):1–14, 2020.

Maryam Negahbani and Deeparnab Chakrabarty. Better algorithms for individually fair k-clustering. *Advances in Neural Information Processing Systems*, 34:13340–13351, 2021.

Ali Vakilian and Mustafa Yalciner. Improved approximation algorithms for individually fair clustering. In *International conference on artificial intelligence and statistics*, pp. 8758–8779. PMLR, 2022.

Xiaoliang Wu, Qilong Feng, Ziyun Huang, Jinhui Xu, and Jianxin Wang. New algorithms for distributed fair k-center clustering: Almost accurate as sequential algorithms. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pp. 1938–1946, 2024.

Zhen Zhang, Xiaohong Chen, Limei Liu, Jie Chen, Junyu Huang, and Qilong Feng. Parameterized approximation schemes for fair-range clustering. *Advances in Neural Information Processing Systems*, 37:60192–60211, 2024.