

A Configuration-LP Framework for Connected k -Median Clustering

Kushagra Chatterjee¹ 

Indian Statistical Institute, Kolkata, India

Rojin Rezvan 

Virginia Tech, Blacksburg, VA, USA

Ali Vakilian 

Virginia Tech, Blacksburg, VA, USA

Abstract

We study the *connected k -median* clustering problem, a clustering problem that augments the classical k -median objective with connectivity constraints. We focus on the *overlapping* variant of the problem, where clusters are allowed to share vertices. In addition to a metric space (V, d) , the input contains a connected graph G on the same vertex set V of size n . The goal is to select at most k centers C and assign vertices to them so as to minimize the k -median cost (i.e., $\sum_{v \in V} d(v, C)$), subject to the constraint that each cluster induces a connected subgraph of G . Since the metric space and the connectivity graph are independent, the problem is significantly more challenging than standard clustering. Eube et al. [29] showed that even the assignment version is $\Omega(\log n)$ -hard to approximate and gave approximation algorithms with guarantees depending polynomially on k .

We develop a configuration-LP-based framework that combines covering LP techniques with a rooted minimum-density oracle. For the assignment version, we obtain an $O(\log^2 n)$ -approximation. For the general version, we develop a bicriteria framework that opens $O(k \log n)$ centers while achieving an $O(\log^2 n)$ -approximation in cost.

2012 ACM Subject Classification Theory of computation $\rightarrow$ Graph algorithms analysis; Theory of computation $\rightarrow$ Facility location and clustering

Keywords and phrases Connected Clustering, k -Median Clustering, Configuration LP

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2026.22

Category APPROX

Funding Ali Vakilian: Supported in part by NSF CAREER Award CCF-2542925.

1 Introduction

Clustering is a central problem in algorithms and data analysis, with classical objectives such as k -center [35, 39] and k -median [15, 14, 19] extensively studied over the past decades. In the k -median problem, we are given a vertex set V of size n together with a metric $d : V \times V \rightarrow \mathbb{R}_{\geq 0}$ and an integer k , and the goal is to select k centers so as to minimize the sum of distances of each vertex to its assigned center. While the vanilla k -median problem admits constant-factor approximation algorithms [15, 14, 5, 23], many modern applications require incorporating additional structural constraints into the clustering process.

In this work, we study the *connected k -median* problem, an extension of the classical k -median problem that incorporates *connectivity constraints*. In addition to the metric space (V, d) , the input includes an unweighted undirected graph $G = (V, E)$, referred to as the *connectivity graph*. The goal is to select k centers and assign vertices to them so as

¹ This work was carried out while the author was a PhD student at the National University of Singapore.



to minimize the k -median objective, subject to the constraint that each cluster induces a connected subgraph in G . Importantly, the graph G is completely independent of the metric d , making the problem fundamentally different from classical clustering.

We focus on the *overlapping* variant of the connected k -median problem, where a vertex may belong to more than one cluster. This is natural in applications such as community detection [8, 30], where individuals can participate in multiple groups, and it also allows vertices such as articulation points to serve as connectors between several clusters. Theoretically, this distinction is crucial; Eube et al. [29] showed that the disjoint version is extremely hard to approximate, admitting an $\Omega(n^{1-\epsilon})$ lower bound even for $k = 2$.

Connected clustering arises naturally in settings where both *attribute data* and *relational structure* are present. For instance, in applications such as community detection in social networks [8, 30] or spatial partitioning tasks like school redistricting [18], the input consists of data points equipped with features as well as an underlying graph capturing relationships such as social interactions or geographic adjacency. In such settings, it is desirable to form clusters that are not only metrically coherent but also respect the connectivity structure of the graph. For example, in school redistricting, geographical units must be grouped into contiguous regions, ensuring that each cluster forms a connected area, while still optimizing criteria such as proximity or compactness. This motivates the study of clustering models that integrate geometric objectives with graph-theoretic constraints.

From an algorithmic perspective, connectivity constraints significantly increase the difficulty of clustering problems. Even in the *assignment version* of connected k -median, where the set of centers is fixed, and the goal is only to assign vertices to centers, Eube et al. [29] showed that the problem is $\Omega(\log n)$ -hard to approximate, and provided an $O(k \log n)$ -approximation algorithm. For the general version, where centers must also be selected, they obtained an $O(k^2 \log n)$ -approximation. These results demonstrate a substantial gap between k -median clustering and its connected variant.

Our contributions. Our main contribution is a configuration-LP framework for connected k -median clustering. Unlike the cut-based relaxation of Eube et al. [29], where connectivity is enforced indirectly through constraints, our LP uses variables indexed by connected sets rooted at centers, thereby encoding connectivity directly into the variables themselves. We combine this formulation with covering-LP techniques and rooted minimum-density oracles to obtain approximation guarantees for connected k -median. Our main results are summarized below.

- **Assignment version.** We obtain an $O(\log^2 n)$ -approximation algorithm for general graphs, which improves Eube et al [29] assignment version result when $k = \omega(\log n)$.
- **General version (bicriteria approximation).** We develop a bicriteria framework that allows opening more than k centers. In particular, we compute a solution that opens at most $O(k \log n)$ centers while achieving an $O(\log^2 n)$ -approximation in cost, yielding an $(O(\log n), O(\log^2 n))$ bicriteria approximation.

The bicriteria guarantee highlights a tradeoff between the number of opened centers and the approximation factor in cost. The algorithm of Eube et al. [29] gives a solution with k centers with approximation factor of $O(k^2 \log n)$ for the general version. In contrast, our bicriteria framework opens $O(k \log n)$ centers and achieves an $O(\log^2 n)$ cost approximation, eliminating the polynomial dependence on k in the objective guarantee.

1.1 Technical Overview

Our algorithms are based on a configuration-LP viewpoint. Instead of assigning vertices to centers directly, we introduce a variable for each pair (c, T) , where c is a center and T is a connected vertex set containing c . Choosing such a configuration corresponds to opening a connected cluster rooted at c . This formulation captures the connectivity requirement directly: every integral solution selects connected sets, and the covering constraints ensure that every vertex is contained in at least one selected cluster. It is useful to contrast this formulation with the cut-based LP used by Eube et al. [29]. In the cut-LP, the variables x_v^c represent fractional assignments of vertices to centers, and connectivity is enforced indirectly through cut constraints: if v is assigned to c , then every v - c cut must contain enough mass assigned to c . This is a natural and compact relaxation, but it is still a *vertex-assignment* formulation; after rounding, one must still ensure that the assigned vertices form connected clusters. In contrast, the configuration LP works directly with connected sets. A variable $y_{T,c}$ corresponds to selecting an entire connected cluster T rooted at c , so connectivity is built into the variables themselves rather than enforced only through constraints.

This global representation is particularly useful for rounding. Once a configuration (c, T) is sampled, it already defines a connected set containing its center, and the union of sampled configurations with the same root remains connected. Thus, the randomized rounding step only needs to ensure coverage of vertices; connectivity is preserved automatically. Moreover, every configuration-LP solution induces a feasible solution to the cut-LP by setting $x_v^c = \sum_{\substack{T \in \mathcal{T}_c \\ v \in T}} y_{T,c}$, with the same objective value, so the configuration LP can be viewed as a more structured relaxation. The price of this stronger representation is that the LP contains exponentially many variables, so solving it exactly is not straightforwardly achievable in polynomial time. In fact, we can only solve this LP relaxation approximately, using column generation based on the rooted minimum-density connected-set oracle.

For the assignment version, where the center set C is fixed, the configuration LP is a pure covering LP. We solve it approximately using the multiplicative weight update framework of Plotkin, Shmoys and Tardos [43] for covering linear programs. The key step is the pricing problem: given weights α_v on the vertices that correspond to the covering constraints, we need to find a connected set $T \ni c$ minimizing the ratio between its cost and the total weight of vertices it covers. This is exactly a rooted minimum-density connected-set problem:

$$\min_{T|T \ni c} \frac{\sum_{v \in T} d(v, c)}{\sum_{v \in T} \alpha_v}.$$

Thus, a ρ -approximate oracle for rooted minimum-density connected sets yields a ρ -approximate pricing oracle for the configuration LP. Combining this with the Plotkin–Shmoys–Tardos framework gives a polynomial-support fractional solution whose cost is within a factor $O(\rho)$ of the optimum LP fractional value. We obtain such density oracles using connections to the *node-weighted Steiner tree* problem.

The rounding step is simple and uses the standard independent randomized rounding. Given a fractional solution $\hat{y}$, we sample each configuration (c, T) with probability proportional to $\hat{y}_{T,c} \log n$. For each center c , the final cluster is the union of all the sampled configurations rooted at c . This union remains connected because every sampled set contains the common root c . The logarithmic oversampling ensures, by a union bound over all vertices, that every vertex is covered with high probability. Linearity of expectation bounds the expected cost by an additional $O(\log n)$ factor. This yields an $O(\rho \log n)$ -approximation for the assignment version. For general graphs, the rooted minimum-density problem admits an $O(\log n)$ -approximation via known algorithms for the *budgeted node-weighted Steiner tree* problem.

This yields an $O(\log^2 n)$ -approximation for the assignment version. For minor-closed graphs, the techniques of [25, 17] suggest that the rooted minimum-density problem may admit an $O(1)$ -approximation, which would improve the assignment-version guarantee to $O(\log n)$.

For the general version, we provide a bicriteria approximation guarantee. We avoid the center-rounding step of Eube et al [29] and instead formulate a non-fixed-center configuration LP. This LP has variables for all roots $c \in V$ and all connected sets $T \in \mathcal{T}_c$, together with a fractional center-budget constraint $\sum_{c \in V} \sum_{T \in \mathcal{T}_c} y_{T,c} \leq k$. This LP is a mixed packing/covering program. Its pricing problem again reduces to a rooted minimum-density connected-set problem, except that the ratio includes both the connection cost of the configuration and the multiplier corresponding to the center-budget constraint. After computing a near-optimal fractional solution, we apply the same randomized rounding scheme. The covering constraints guarantee that all vertices are covered with high probability, while the budget constraint implies that the number of sampled configurations, and hence the number of opened centers, is at most $O(k \log n)$ with high probability. This gives an $(O(\log n), O(\rho \log n))$ bicriteria approximation, which becomes $(O(\log n), O(\log^2 n))$ for general graphs.

1.2 Related Work

Clustering problems such as k -center and k -median are among the most well-studied optimization problems in algorithm design. Both problems are known to be NP-hard. For the k -center problem, a tight 2-approximation algorithm is known [33, 35], and this factor is optimal unless $P=NP$ [36]. For metric k -median, the current best polynomial-time approximation guarantee is $2 + \varepsilon$, for any fixed $\varepsilon > 0$, achieved by the recent algorithm of Cohen-Addad et al. [22]. In contrast, no polynomial-time algorithm can achieve an approximation ratio smaller than $1 + 2/e \approx 1.736$, unless $P = NP$ [37].

Connected clustering. Clustering with connectivity constraints has been studied in several contexts. Ge et al. [31] introduced the connected k -center problem as a model for jointly clustering attribute data and relationship data: the metric captures similarity between attribute vectors, while the connectivity graph captures relational information. They gave an exact dynamic program when the connectivity graph is a tree, and also studied general graphs, although without a known approximation guarantee. For general graphs with disjoint clusters, Drexler et al. [26] obtained an $O(\log^2 k)$ -approximation; they also gave constant-factor approximations for Euclidean metrics of constant dimension and metrics of constant doubling dimension, motivated by applications in geodesy and sea-level analysis. In contrast, when overlapping clusters are allowed, connected k -center admits a simple 2-approximation, highlighting an important distinction between the disjoint and overlapping models.

For the k -median objective, the connected variant has been studied more recently. Validi et al. [46] consider connected k -median in the context of districting, where geographical units must be grouped into contiguous regions while optimizing a proximity-based objective. They formulate the problem as an integer linear program, which can be solved optimally but requires exponential time in the worst case. Gupta et al. [34] study the inapproximability of clustering with connectivity constraints and show that even when $k = 2$ and the connectivity graph is star-like, the problem is $\Omega(\log n)$ -hard to approximate. They also provide an $O(\log n)$ -approximation algorithm for some restricted instances of the problem.

Clustering with additional constraints. Beyond connectivity, a large body of work has focused on incorporating additional constraints into clustering problems motivated by real-world applications. Capacitated clustering introduces upper bounds on cluster sizes [40], while

models with outliers allow for excluding a small fraction of points from the clustering [16]. More recently, fairness constraints have been extensively studied, requiring clusters to satisfy various fairness conditions [21, 9, 6, 10, 24, 38, 41, 45, 32, 1, 42, 28, 27, 20]. Fairness constraints have also been studied for other types of clustering problems, like correlation clustering [3, 2, 4] and consensus clustering [13, 11, 12]. These variants highlight the increasing importance of structured clustering models, where combinatorial constraints are integrated with geometric objectives.

2 Preliminaries and Problem Statement

In this section, we introduce the definitions and notation used throughout the paper. We also formally state the problem and present our main results.

Problem Statement. We are given a vertex set V , a metric $d : V \times V \rightarrow \mathbb{R}_{\geq 0}$, and a connected *connectivity graph* $G = (V, E)$, which is unrelated to d . We consider the problem in two settings.

- **Assignment Version:** A set of centers $C \subseteq V$ with $|C| \leq k$, for some integer k is given. A feasible *overlapping* clustering solution outputs connected sets $\mathcal{S} := (S_c)_{c \in C}$ such that (i) $c \in S_c$ for all $c \in C$, (ii) $G[S_c]$ is connected for all $c \in C$, and (iii) $\bigcup_{c \in C} S_c = V$.
- **General Version:** An integer k is given, and the goal is to choose a set of centers $C \subseteq V$ with $|C| \leq k$ together with connected sets $\mathcal{S} := (S_c)_{c \in C}$ satisfying the same conditions as above.

In both cases, the objective is to find a feasible solution $\mathcal{S}$ that minimizes

$$\text{cost}(\mathcal{S}) := \sum_{c \in C} \sum_{v \in S_c} d(v, c),$$

where a vertex assigned to multiple clusters contributes once per cluster. Here $G[S_c]$ denotes the subgraph of G induced by the vertex set S_c .

► **Remark 1.** Let $G_1, \dots, G_\ell$ be the connected components of G . In the assignment version, if some component contains at least one vertex but no center, then the instance is infeasible. Otherwise, each component can be solved independently, and the solutions can be combined. Hence, without loss of generality, we assume throughout that G is connected and every vertex can reach some center.

► **Remark 2 (Bounded aspect ratio).** Throughout the paper we assume the metric d has polynomially bounded aspect ratio: if $D_{\max} := \max_{u, v \in V} d(u, v)$ and $\delta_{\min} := \min_{u \neq v \in V} d(u, v)$, we assume

$$\Delta := \frac{D_{\max}}{\delta_{\min}} = \text{poly}(n).$$

This is the standard bounded-spread condition used throughout the metric clustering and embedding literature. All stated polynomial running times in this paper are with respect to n , k , and $1/\varepsilon$ under this assumption.

Our Results. We now formally state our results. A feasible overlapping clustering solution for either version is referred to as a *connected k -median clustering* of the vertex set V . We say that a clustering $\mathcal{S}$ is an α -approximate connected k -median clustering if $\text{cost}(\mathcal{S}) \leq \alpha \cdot \text{cost}(\mathcal{S}^*)$, where $\mathcal{S}^*$ denotes an optimal connected k -median clustering.

For the assignment version, we obtain an improved approximation guarantee when $k = \omega(\log n)$. In particular, we show the following.

► **Theorem 3.** *Given a vertex set V , a metric $d : V \times V \rightarrow \mathbb{R}_{\geq 0}$, an undirected, unweighted connectivity graph $G = (V, E)$, and a set of centers $C \subseteq V$ with $|C| \leq k$, there exists a polynomial-time algorithm that computes an $O(\log^2 n)$ -approximate connected k -median clustering.*

We now consider the general version of the problem, where the set of centers is not given and must be chosen as part of the solution. For the general version, we have obtained bicriteria approximation guarantees. For this, we need to define a bicriteria notion of approximation that allows the algorithm to open more than k centers. A clustering $\mathcal{S}$ is said to be a (β, α) -approximate connected k -median clustering if it uses at most βk centers and satisfies $\text{cost}(\mathcal{S}) \leq \alpha \cdot \text{cost}(\mathcal{S}^*)$, where $\mathcal{S}^*$ is an optimal solution that uses at most k centers.

We obtain the following bicriteria guarantees.

► **Theorem 4.** *Given a vertex set V , a metric $d : V \times V \rightarrow \mathbb{R}_{\geq 0}$, an undirected, unweighted connectivity graph $G = (V, E)$ and an integer k , there exists a polynomial-time algorithm that computes an $(O(\log n), O(\log^2 n))$ -approximate connected k -median clustering.*

3 Assignment Version

In this section, we prove our results for the assignment version of the problem. In particular, we prove Theorem 3, which we restate below.

► **Theorem 3.** *Given a vertex set V , a metric $d : V \times V \rightarrow \mathbb{R}_{\geq 0}$, an undirected, unweighted connectivity graph $G = (V, E)$, and a set of centers $C \subseteq V$ with $|C| \leq k$, there exists a polynomial-time algorithm that computes an $O(\log^2 n)$ -approximate connected k -median clustering.*

To prove Theorem 3, we employ a *configuration LP* formulation. Since this LP has exponentially many variables, we compute a near-optimal fractional solution using a *rooted minimum-density connected set oracle* in conjunction with the Plotkin–Shmoys–Tardos [43] multiplicative weight update framework for covering LPs. We then round this fractional solution to obtain the desired approximation guarantee. The details are presented in the subsequent sections.

3.1 Configuration LP

We follow a *configuration LP* route: formulate an exponential-size covering LP over all connected clusters, solve it approximately via column generation/packing-covering with a *minimum-density* oracle, and round it in one shot via randomized rounding. This yields an $O(\rho \log n)$ approximation given a ρ -approximation oracle for the rooted minimum-density connected set problem.

For each center $c \in C$, let $\mathcal{T}_c$ denote the family of all connected vertex sets $T \subseteq V$ such that $c \in T$ and $G[T]$ is connected. For $T \in \mathcal{T}_c$, define

$$\text{cost}_c(T) := \sum_{v \in T} d(v, c).$$

We introduce a variable $y_{T,c} \geq 0$ for each pair (c, T) to represent the extent to which we choose configuration (c, T) . We define the primal and dual relaxation of the assignment problem as follows.

Primal (Configuration) LP

$$\min \sum_{c \in C} \sum_{T \in \mathcal{T}_c} \text{cost}_c(T) y_{T,c} \quad (1)$$

$$\text{s.t.} \quad \sum_{c \in C} \sum_{\substack{T \in \mathcal{T}_c \\ v \in T}} y_{T,c} \geq 1 \quad \forall v \in V \setminus C, \quad (2)$$

$$y_{T,c} \geq 0 \quad \forall c \in C, \forall T \in \mathcal{T}_c. \quad (3)$$

Dual LP

$$\max \sum_{v \in V \setminus C} \alpha_v \quad (4)$$

$$\text{s.t.} \quad \sum_{v \in T \setminus \{c\}} \alpha_v \leq \text{cost}_c(T) \quad \forall c \in C, \forall T \in \mathcal{T}_c, \quad (5)$$

$$\alpha_v \geq 0 \quad \forall v \in V \setminus C. \quad (6)$$

► **Lemma 5.** *The configuration LP (1)–(3) is a valid relaxation of the assignment version of connected k -median with overlapping clusters. In particular, its optimal value is at most OPT.*

Proof. We provide the proof in the appendix. ◀

3.2 The Oracle: Rooted Minimum-Density Connected Set

Since the configuration LP has exponentially many variables (one for each connected set $T \in \mathcal{T}_c$ and each center $c \in C$), it is not immediate how to solve it in polynomial time. We follow the standard oracle-based approach for large-scale packing/covering linear programs: rather than enumerating all configurations explicitly, we access them through an optimization subroutine that, for suitable dual weights, returns an approximately best column [43, 47]. In our setting, this subroutine is the *rooted minimum-density connected-set problem* defined below.

► **Definition 6.** *Given a graph $G = (V, E)$, a root $c \in V$, nonnegative node profits $\alpha : V \rightarrow \mathbb{R}_{\geq 0}$, and nonnegative node costs $w : V \rightarrow \mathbb{R}_{\geq 0}$, the rooted minimum-density problem asks for a connected set $T \ni c$ minimizing*

$$\text{dens}(T) := \frac{\sum_{v \in T} w(v)}{\sum_{v \in T} \alpha_v},$$

with the convention $\text{dens}(T) = +\infty$ if $\sum_{v \in T} \alpha_v = 0$.

In our application, the dual variables are defined only on $V \setminus C$, so we set $\alpha_c := 0$. Thus $\sum_{v \in T} \alpha_v = \sum_{v \in T \setminus \{c\}} \alpha_v$, which matches the left-hand side of the dual packing constraint (5). In particular, minimizing $\text{dens}(T)$ identifies the most violated dual constraint, and hence the most promising column for the primal configuration LP.

► **Definition 7.** *A ρ -approximate density oracle returns a connected set $T \ni c$ such that $\text{dens}(T) \leq \rho \cdot \min_{T' \ni c} \text{dens}(T')$.*

► **Lemma 8.** *There is a polynomial-time $O(\log n)$ -approximate density oracle.*

Proof. The proof uses a result by Bateni et al [7]. We provide the full proof in the appendix. ◀

3.3 Computing a ρ -Approximate Solution of the Assignment Version of the Configuration LP

We next explain how to approximately solve the configuration LP using the rooted minimum-density oracle from the previous subsection. We follow the standard oracle-based approach for large covering LPs: we view the configuration LP as a covering problem over an implicit column set and apply the framework of Plotkin–Shmoys–Tardos [43]; the key point is that the corresponding pricing subproblem is exactly a rooted minimum-density connected-set problem. Therefore, a ρ -approximate density oracle yields, in polynomial time, a $(1 + \varepsilon)\rho$ -approximate fractional solution to the configuration LP.

► **Theorem 9.** *Assume a ρ -approximate density oracle. Then for any $\varepsilon \in (0, 1)$, one can compute in polynomial time a feasible fractional solution $\hat{y}$ to the Configuration LP (1)–(3) satisfying*

$$\sum_{c \in C} \sum_{T \in \mathcal{T}_c} \text{cost}_c(T) \hat{y}_{T,c} \leq (1 + \varepsilon)\rho \cdot \text{OPT}_{\text{CLP}} \leq (1 + \varepsilon)\rho \cdot \text{OPT},$$

with support size $\text{poly}(n, \varepsilon^{-1}, \rho)$.

Proof. We cast the configuration LP as a pure covering LP and apply a standard fractional packing/covering approximation scheme (Plotkin–Shmoys–Tardos [43]), whose guarantee is that an approximately optimal *fractional* solution can be computed given oracle access to an approximate *pricing* subproblem.

We discard all columns (c, T) with $T \cap (V \setminus C) = \emptyset$. Such columns cover no covering constraint and therefore play no role in the covering LP. After this restriction, every remaining column satisfies $\text{cost}_c(T) > 0$.

Step 1: Configuration LP as a covering LP. Let $U_0 := V \setminus C$ and $n_0 := |U_0|$. Index columns by pairs (c, T) with $c \in C$ and $T \in \mathcal{T}_c$. Define the incidence matrix $A \in \{0, 1\}^{n_0 \times N}$ and cost vector $c \in \mathbb{R}_{>0}^N$ by

$$A_{v,j} := \mathbf{1}[v \in T], \quad c_j := \text{cost}_c(T) = \sum_{u \in T} d(u, c).$$

Since d is a metric with $d(u, c) > 0$ for all $u \neq c$, every column $j = (c, T)$ with $T \neq \{c\}$ satisfies $c_j > 0$, so there are no 0/0 anomalies in the pricing step. The configuration LP (1)–(3) is then $\min\{c^\top y : Ay \geq \mathbf{1}, y \geq 0\}$, a non-negative covering LP with $m := n_0$ constraints and N columns.

Step 2: Pricing is (approximate) rooted minimum-density. In the Plotkin–Shmoys–Tardos framework for covering LPs, the algorithm maintains a non-negative weight vector $p \in \mathbb{R}_{\geq 0}^{U_0}$ on constraints and, in each iteration, needs a column whose *cost per covered weight* is minimum (equivalently, whose covered weight per unit cost is maximum). For a column $j = (c, T)$, the covered weight under p is $\langle p, A_{\cdot,j} \rangle = \sum_{v \in T \cap U_0} p_v$, so the ratio is

$$\frac{c_j}{\langle p, A_{\cdot,j} \rangle} = \frac{\sum_{u \in T} d(u, c)}{\sum_{v \in T \cap U_0} p_v}.$$

This is the rooted minimum-density objective (Definition 6) with root c , costs $w_c(u) := d(u, c)$, and profits $\alpha_v := p_v$ (with $\alpha_c := 0$ for centers). A ρ -approximate density oracle therefore provides a ρ -approximate pricing oracle for the covering LP: calling it for each $c \in C$ and returning the column of globally smallest density gives an ρ -approximate minimizer of the ratio over all columns.

Step 3: Width and the upper bound q on OPT_{CLP} . To apply Theorem 10, we need (i) an upper bound $q \geq \text{OPT}_{\text{CLP}}$ and (ii) the associated width

$$W \geq q \cdot \max_{v \in U_0, j=(c,T)} \frac{A_{v,j}}{c_j} = q \cdot \max_{j=(c,T): T \neq \{c\}} \frac{1}{\text{cost}_c(T)}. \quad (7)$$

Upper bound q . Construct the following explicit feasible solution: for each $v \in U_0$, let P_v be a shortest G -path from the nearest center c_v to v , which exists by the assumption on connectivity of G , and set $y_{V(P_v), c_v} = 1$ (with remaining variables 0). This is feasible (every v is covered) and has cost

$$q := \sum_{v \in U_0} \text{cost}_{c_v}(V(P_v)) \leq \sum_{v \in U_0} \sum_{u \in V(P_v)} d(u, c_v) \leq n_0 \cdot n \cdot D_{\max},$$

where $D_{\max} := \max_{u,v \in V} d(u, v)$. Since the “Width W ” step below normalizes so that $\delta_{\min} = 1$, then $D_{\max}$ coincides with the aspect ratio Δ , which is $\text{poly}(n)$ by Remark 2. Therefore, $q = \text{poly}(n)$.

Width W . After normalizing d by its minimum positive value so that $\delta := \min_{u \neq v} d(u, v) = 1$ (this leaves all approximation ratios unchanged and ensures $c_j \geq 1$ for all non-trivial columns), the width in (7) becomes

$$W = q \cdot \max_j \frac{1}{c_j} \leq q \cdot \frac{1}{\delta} = q = \text{poly}(n).$$

Step 4: Apply Theorem 10. With $m = n_0$, the ρ -approximate pricing oracle from Step 2, the upper bound $q \geq \text{OPT}_{\text{CLP}}$ from Step 3, and the width $W \leq q$ from Step 3, Theorem 10 yields a feasible $\hat{y}$ with

$$c^\top \hat{y} \leq \rho(1 + \varepsilon + \varepsilon^2) \text{OPT}_{\text{CLP}} \leq (1 + \varepsilon) \rho \cdot \text{OPT}_{\text{CLP}}$$

absorbing ε^2 into ε by replacing ε with $\varepsilon/2$, and support size

$$|\text{supp}(\hat{y})| = \tilde{O}\left(\frac{n_0 W \rho}{\varepsilon^3}\right) = \tilde{O}\left(\frac{n_0 q \rho}{\varepsilon^3}\right) = \text{poly}(n).$$

Since $\text{OPT}_{\text{CLP}} \leq \text{OPT}$ (Lemma 5), the stated bound $c^\top \hat{y} \leq (1 + \varepsilon) \rho \cdot \text{OPT}$ follows.

Runtime. Each iteration of the Plotkin–Shmoys–Tardos framework calls the density oracle $k := |C|$ times (once per center) and selects the column of globally smallest density. The total number of iterations is $\tilde{O}(n_0 W \rho / \varepsilon^3) = \text{poly}(n)$. After all iterations, the restricted LP on the generated columns is solved exactly in polynomial time. Each oracle call is polynomial by Lemma 8, so the entire algorithm runs in polynomial time. $\blacktriangleleft$

The following theorem is a standard oracle-based reformulation of the fractional covering framework of Plotkin–Shmoys–Tardos; we cite Sharma [44, Theorems 3 and 4] for a clean restatement with approximate pricing and support bounds. For completeness, we provide a self-contained proof of Theorem 10 in the appendix.

► **Theorem 10** (Plotkin–Shmoys–Tardos [43]; cf. Sharma [44, Theorems 3 and 4]). *Consider the covering linear program $\min\{c^\top y : Ay \geq \mathbf{1}, y \geq 0\}$, where $A \in \mathbb{R}_{\geq 0}^{m \times N}$ and $c \in \mathbb{R}_{\geq 0}^N$. Let OPT denote its optimal value, and let $q \geq \text{OPT}$ be a given upper bound on the optimal objective. Assume that for every weight vector $w \in \mathbb{R}_{\geq 0}^m$, there is a polynomial-time α -approximate pricing oracle returning a column $j \in [N]$ with*

$$\frac{c_j}{\sum_i A_{ij} w_i} \leq \alpha \cdot \min_{j' \in [N]} \frac{c_{j'}}{\sum_i A_{ij'} w_i}.$$

Let $W \geq q \max_{i,j} (A_{ij}/c_j)$ be a width bound. Then for every $\varepsilon \in (0, 1]$, there is a polynomial-time algorithm that computes a feasible $\hat{y}$ with $c^\top \hat{y} \leq \alpha(1 + \varepsilon + \varepsilon^2)\text{OPT}$. Moreover, $|\text{supp}(\hat{y})| = \tilde{O}(mW\alpha/\varepsilon^3)$.

► **Remark 11.** The approximation ratio $\alpha(1 + \varepsilon + \varepsilon^2)$ in Theorem 10 depends only on the oracle quality α and the accuracy parameter ε ; it is *independent* of the width W . The width W affects only the support size and iteration count, not the approximation guarantee. In our setting $W \leq q = \text{poly}(n)$, so both quantities are polynomial.

3.4 Rounding the Configuration LP

► **Theorem 12.** Let $\hat{y}$ be any feasible solution to (1)–(3) with polynomial support. There is a randomized polynomial-time algorithm that, with high probability, outputs a feasible integral overlap solution $(S_c)_{c \in C}$ covering all of $V \setminus C$ and its cost is $O(\log n) \cdot \sum_{c \in C} \sum_{T \in \mathcal{T}_c} \text{cost}_c(T) \hat{y}_{T,c}$.

Proof. Fix $\lambda \geq 2$. Independently for each pair (c, T) in the support of $\hat{y}$ (which is of polynomial size), sample T with probability $p_{T,c} := \min\{1, \lambda \ln(n) \cdot \hat{y}_{T,c}\}$. For each $c \in C$, let $\mathcal{F}_c$ be the (multi)set of the sampled configurations and define $S_c := \{c\} \cup \bigcup_{T \in \mathcal{F}_c} T$. Adding $\{c\}$ ensures $c \in S_c$ even if $\mathcal{F}_c = \emptyset$; since $d(c, c) = 0$, this adds zero cost. Each sampled T is connected and contains c , so S_c is a union of connected sets sharing c , hence connected.

Coverage. Fix $v \in V \setminus C$. Feasibility of $\hat{y}$ gives $\sum_{c,T:v \in T} \hat{y}_{T,c} \geq 1$. If any $p_{T,c} = 1$ for a pair with $v \in T$, then v is certainly covered. Otherwise $p_{T,c} = \lambda \ln(n) \hat{y}_{T,c}$ for all covering pairs, and

$$\Pr[v \text{ uncovered}] = \prod_{c,T:v \in T} (1 - p_{T,c}) \leq \exp\left(-\sum_{c,T:v \in T} p_{T,c}\right) \leq \exp(-\lambda \ln n_0) = n_0^{-\lambda}.$$

Union bound over $v \in V \setminus C$ (there are n_0 such vertices):

$$\Pr[\text{any } v \text{ uncovered}] \leq n_0 \cdot n_0^{-\lambda} \leq n_0^{-(\lambda-1)} \leq 1/n_0 = 1/\Omega(n)$$

for $\lambda \geq 2$.

Cost. For each c , since the cost of a union is at most the sum of costs:

$$\sum_{v \in S_c} d(v, c) \leq \sum_{T \in \mathcal{F}_c} \sum_{v \in T} d(v, c) = \sum_T \mathbf{1}[T \text{ sampled}] \cdot \text{cost}_c(T).$$

Taking expectation and summing over c :

$$\mathbb{E}[\text{ALG}] \leq \sum_{c,T} \text{cost}_c(T) \cdot p_{T,c} \leq \lambda \ln(n) \sum_{c,T} \text{cost}_c(T) \hat{y}_{T,c} = O(\log n) \sum_{c,T} \text{cost}_c(T) \hat{y}_{T,c}.$$

By Markov's inequality, a single run has cost at most twice its expectation with constant probability; therefore, by performing $O(\log n)$ independent runs and returning the minimum-cost feasible solution, we obtain an $O(\log n)$ -approximation with high probability. ◀

3.5 Proof of Theorem 3

Now we are ready to complete the proof of Theorem 3.

Proof of Theorem 3. By combining Theorem 9 and Theorem 12, we obtain that if there exists a ρ -approximate density oracle, then there is a polynomial-time algorithm that computes an $O(\rho \log n)$ -approximate connected k -median clustering. For general graphs, Lemma 8 provides an $O(\log n)$ -approximate density oracle. Substituting $\rho = O(\log n)$, we obtain an $O(\log^2 n)$ -approximation. ◀

4 General Version

In this section, we establish bicriteria approximation guarantees for the general version of the connected k -median clustering problem. In particular, we prove Theorem 4, which we restate below.

► **Theorem 4.** *Given a vertex set V , a metric $d : V \times V \rightarrow \mathbb{R}_{\geq 0}$, an undirected, unweighted connectivity graph $G = (V, E)$ and an integer k , there exists a polynomial-time algorithm that computes an $(O(\log n), O(\log^2 n))$ -approximate connected k -median clustering.*

To prove Theorem 4, we follow an approach similar to that used for the assignment version. We first formulate a configuration LP for the general case and compute a near-optimal fractional solution in polynomial time. We then round this solution to obtain the desired bicriteria guarantee. We provide the details in the subsequent sections.

4.1 General Version of the Configuration LP

Primal (Configuration) LP

$$\min \sum_{c \in V} \sum_{T \in \mathcal{T}_c} \text{cost}_c(T) y_{T,c} \quad (8)$$

$$\text{s.t.} \quad \sum_{c \in V} \sum_{\substack{T \in \mathcal{T}_c \\ v \in T}} y_{T,c} \geq 1 \quad \forall v \in V, \quad (9)$$

$$\sum_{c \in V} \sum_{T \in \mathcal{T}_c} y_{T,c} \leq k \quad (10)$$

$$y_{T,c} \geq 0 \quad \forall c \in V, \forall T \in \mathcal{T}_c. \quad (11)$$

4.2 Computing a ρ -Approximate Solution of the Configuration LP for the General Variant

First, we interpret the center-budget constraint as $\sum_{c \in V} \sum_{T \in \mathcal{T}_c} y_{T,c} \leq k$, which bounds the total configuration mass. This serves as a fractional relaxation of the requirement that at most k centers are opened. In particular, the variables $y_{T,c}$ are now defined for all $c \in V$ and all $T \in \mathcal{T}_c$. Let OPT_{GCLP} denote the optimal value of this general configuration LP.

► **Lemma 13.** *Assume a ρ -approximate rooted minimum-density oracle for arbitrary non-negative node costs and profits. Then, for every $\varepsilon \in (0, 1)$, one can compute in polynomial time a polynomial-support fractional solution $\hat{y}$ satisfying*

$$\sum_{c \in V} \sum_{\substack{T \in \mathcal{T}_c \\ v \in T}} \hat{y}_{T,c} \geq 1 \quad \forall v \in V, \quad (12)$$

$$\sum_{c \in V} \sum_{T \in \mathcal{T}_c} \hat{y}_{T,c} \leq (1 + \varepsilon)k, \quad (13)$$

$$\sum_{c \in V} \sum_{T \in \mathcal{T}_c} \text{cost}_c(T) \hat{y}_{T,c} \leq (1 + \varepsilon)\rho \cdot \text{OPT}_{\text{GCLP}}. \quad (14)$$

Moreover, $\hat{y}$ may be chosen with support size polynomial in the input size and $1/\varepsilon$.

Proof. Write the general configuration LP as a mixed packing/covering program with one packing constraint. The covering constraints are indexed by vertices $v \in V$, and the columns are indexed by pairs (c, T) with $c \in V$ and $T \in \mathcal{T}_c$. For such a column, define

$$A_{v,(c,T)} := \mathbf{1}[v \in T], \quad b_{(c,T)} := 1, \quad q_{(c,T)} := \text{cost}_c(T).$$

Thus the LP has the form

$$\min \sum_{c,T} q_{(c,T)} y_{T,c} \quad \text{s.t.} \quad Ay \geq \mathbf{1}, \quad \sum_{c,T} b_{(c,T)} y_{T,c} \leq k, \quad y \geq 0.$$

We apply the standard oracle-based framework for mixed packing and covering programs, as in Young [47] and Plotkin–Shmoys–Tardos [43]. It suffices to identify the pricing problem required by this framework.

For a guessed objective value τ , the feasibility version is

$$Ay \geq \mathbf{1}, \quad \sum_{c,T} y_{T,c} \leq k, \quad \sum_{c,T} \text{cost}_c(T) y_{T,c} \leq \tau, \quad y \geq 0.$$

In the mixed packing/covering framework, a pricing step is given nonnegative weights p_v on the covering constraints and nonnegative multipliers η, θ for the two packing constraints. It asks for a column minimizing the ratio

$$\frac{\eta \cdot b_{(c,T)} + \theta \cdot q_{(c,T)}}{\sum_{v \in V} p_v A_{v,(c,T)}} = \frac{\eta + \theta \text{cost}_c(T)}{\sum_{v \in T} p_v}.$$

For a fixed root c , this is exactly a rooted minimum-density connected-set problem. Indeed, define node costs $w_c^{\eta,\theta}(u) := \theta d(u, c) + \eta \cdot \mathbf{1}[u = c]$, and node profits $\alpha_u := p_u$. Since every $T \in \mathcal{T}_c$ contains c , we have

$$\sum_{u \in T} w_c^{\eta,\theta}(u) = \theta \sum_{u \in T} d(u, c) + \eta = \theta \text{cost}_c(T) + \eta.$$

Therefore the pricing ratio is precisely $\frac{\sum_{u \in T} w_c^{\eta,\theta}(u)}{\sum_{v \in T} \alpha_v}$. Calling the rooted minimum-density oracle once for each possible root $c \in V$ and taking the best returned column gives a ρ -approximate pricing oracle for the mixed packing/covering instance.

By the standard guarantee for mixed packing and covering with a ρ -approximate pricing oracle, binary search over τ and the above pricing routine compute, in polynomial time, a solution satisfying (12), (13), and (14). The factor $(1 + \varepsilon)$ accounts for the usual accuracy loss in the mixed packing/covering feasibility procedure.

Finally, the algorithm generates only polynomially many columns. Solving the restricted mixed LP over the generated columns and taking a basic feasible solution yields a polynomial-support solution $\hat{y}$ with the same bounds. $\blacktriangleleft$

► **Corollary 14.** *For general connectivity graphs, one can compute in polynomial time a fractional solution $\hat{y}$ satisfying (12), (13), and*

$$\sum_{c \in V} \sum_{T \in \mathcal{T}_c} \text{cost}_c(T) \hat{y}_{T,c} \leq O(\log n) \cdot \text{OPT}_{\text{GCLP}}.$$

Proof. By Lemma 8, the rooted minimum-density oracle has approximation ratio $\rho = O(\log n)$ on general graphs. Applying Lemma 13 with a constant ε gives the claim. $\blacktriangleleft$

4.3 Rounding the General Version of the Configuration LP

In this subsection, we round the fractional solution obtained from Lemma 13.

► **Lemma 15.** *Suppose we are given a polynomial-support feasible solution $\hat{y}$ to (8)–(11) of cost at most Γ . Then there is a randomized polynomial-time algorithm that outputs, with high probability, an overlapping connected clustering using at most $O(k \log n)$ centers of cost $O(\log n) \cdot \Gamma$.*

Proof. Fix a constant $\lambda \geq 2$. Independently for every pair (c, T) in the support of $\hat{y}$, sample the configuration T with probability $p_{T,c} := \min\{1, \lambda \ln n \cdot \hat{y}_{T,c}\}$. For each $c \in V$, let $\mathcal{F}_c$ be the collection of the sampled configurations rooted at c . We open center c if $\mathcal{F}_c \neq \emptyset$, and define its cluster as $S_c := \bigcup_{T \in \mathcal{F}_c} T$. Since every sampled set $T \in \mathcal{T}_c$ contains c and is connected, the union S_c is connected whenever $\mathcal{F}_c \neq \emptyset$.

We first prove coverage. Fix a vertex $v \in V$. By feasibility of $\hat{y}$, $\sum_{c \in V} \sum_{\substack{T \in \mathcal{T}_c \\ v \in T}} \hat{y}_{T,c} \geq 1$. If there exists a pair (c, T) with $v \in T$ and $p_{T,c} = 1$, then v is covered with probability one. Otherwise,

$$\begin{aligned} \Pr[v \text{ is uncovered}] &= \prod_{c, T: v \in T} (1 - p_{T,c}) \leq \exp\left(-\sum_{c, T: v \in T} p_{T,c}\right) \\ &= \exp\left(-\lambda \ln n \sum_{c, T: v \in T} \hat{y}_{T,c}\right) \leq n^{-\lambda}. \end{aligned}$$

Taking a union bound over all vertices, the probability that some vertex is uncovered is at most $n^{1-\lambda}$, which is at most $1/n$ for $\lambda \geq 2$.

Next, we bound the number of opened centers. Let N be the number of the sampled configurations. Since the number of opened centers is at most N , it suffices to bound N . We have

$$\mathbb{E}[N] = \sum_{c \in V} \sum_{T \in \mathcal{T}_c} p_{T,c} \leq \lambda \ln n \sum_{c \in V} \sum_{T \in \mathcal{T}_c} \hat{y}_{T,c} \leq \lambda(1 + \varepsilon)k \ln n,$$

where the last inequality follows from Constraint (13). By a Chernoff bound, $N = O(k \log n)$ with high probability. Therefore, the algorithm opens at most $O(k \log n)$ centers with high probability.

Finally, we bound the cost. Since the cost of a union is at most the sum of the costs of the sampled configurations,

$$\text{cost}(\{S_c\}) = \sum_{c: \mathcal{F}_c \neq \emptyset} \sum_{v \in S_c} d(v, c) \leq \sum_{c \in V} \sum_{T \in \mathcal{F}_c} \text{cost}_c(T).$$

Taking expectation,

$$\mathbb{E}[\text{cost}(\{S_c\})] \leq \sum_{c \in V} \sum_{T \in \mathcal{T}_c} p_{T,c} \text{cost}_c(T) \leq \lambda \ln n \sum_{c \in V} \sum_{T \in \mathcal{T}_c} \text{cost}_c(T) \hat{y}_{T,c} = O(\log n) \cdot \Gamma.$$

By Markov's inequality, a single run has cost at most twice its expectation with constant probability; therefore, by performing $O(\log n)$ independent runs and returning the minimum-cost feasible solution, we obtain an $O(\log n)$ -approximation with high probability. This proves the claimed bicriteria guarantee. ◀

4.4 Proof of Theorem 4

Now, we are ready to complete the proof of Theorem 4.

Proof of Theorem 4. Let $\hat{y}$ be the fractional solution returned by Lemma 13 with a fixed constant $\varepsilon \in (0, 1)$. Then $\hat{y}$ satisfies the covering constraints and $\sum_{c \in V} \sum_{T \in \mathcal{T}_c} \hat{y}_{T,c} \leq (1 + \varepsilon)k$, and its cost is at most $(1 + \varepsilon)\rho \cdot \text{OPT}_{\text{GCLP}} \leq (1 + \varepsilon)\rho \cdot \text{OPT}$, where the last inequality follows because the general configuration LP is a relaxation of the non-assignment overlapping connected k -median problem.

Now apply Lemma 15 to $\hat{y}$ with $\Gamma := (1 + \varepsilon)\rho \cdot \text{OPT}$. The rounding opens at most $O((1 + \varepsilon)k \log n) = O(k \log n)$ centers with high probability, and its expected cost is at most $O(\log n) \cdot \Gamma = O(\rho \log n) \cdot \text{OPT}$. Thus we obtain an $(O(\log n), O(\rho \log n))$ bicriteria approximation.

For general graphs, Lemma 8 gives $\rho = O(\log n)$. Therefore the bicriteria guarantee becomes $(O(\log n), O(\log^2 n))$. This proves Theorem 4. $\blacktriangleleft$

5 Conclusion and Open Problems

In this paper, we developed a configuration-LP framework for the connected k -median problem. Unlike previous cut-based relaxations, our formulation incorporates connectivity directly into the LP variables by working with connected configurations rooted at centers. We combined this viewpoint with covering-LP techniques and rooted minimum-density oracles to obtain approximation guarantees for both the assignment version and a bicriteria variant of the general version. In particular, we obtained an $O(\log^2 n)$ -approximation for the assignment version on general graphs, together with an $(O(\log n), O(\log^2 n))$ bicriteria approximation for the general version.

More broadly, our results suggest that configuration LPs provide a natural framework for handling connectivity constraints in clustering problems. By separating the problem into a covering-LP layer and a graph-theoretic rooted density oracle, the framework makes it possible to incorporate stronger oracles for special graph families and related connectivity models. We believe this viewpoint may be useful beyond connected k -median, particularly for other constrained clustering objectives where connectivity plays a central role.

Several interesting open problems remain. The most natural question is whether one can obtain polylogarithmic approximation guarantees for the general version containing exactly k centers. More generally, it would be interesting to better understand the power and limitations of configuration-LP formulations for connectivity-constrained clustering problems. Another promising direction is to design stronger rooted minimum-density oracles for restricted graph families, such as minor-closed graphs. Since the rooted minimum-density problem already admits an $O(\log n)$ -approximation in general graphs, improving this guarantee on structured graph families would immediately translate into stronger approximation guarantees within our framework.

References

- 1 Mohsen Abbasi, Aditya Bhaskara, and Suresh Venkatasubramanian. Fair clustering via equitable group representations. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 504–514, 2021. doi:10.1145/3442188.3445913.
- 2 Saba Ahmadi, Sainyam Galhotra, Barna Saha, and Roy Schwartz. Fair correlation clustering. *arXiv preprint arXiv:2002.03508*, 2020. arXiv:2002.03508.

- 3 Sara Ahmadian, Alessandro Epasto, Ravi Kumar, and Mohammad Mahdian. Fair correlation clustering. In *International conference on artificial intelligence and statistics*, pages 4195–4205. PMLR, 2020. URL: <http://proceedings.mlr.press/v108/ahmadian20a.html>.
- 4 Sara Ahmadian and Maryam Negahbani. Improved approximation for fair correlation clustering. In *International Conference and Statistics*, pages 9499–9516. PMLR, 2023. URL: <https://proceedings.mlr.press/v206/ahmadian23a.html>.
- 5 Vijay Arya, Naveen Garg, Rohit Khandekar, Adam Meyerson, Kamesh Munagala, and Vinayaka Pandit. Local search heuristic for k-median and facility location problems. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 21–29, 2001. doi:10.1145/380752.380755.
- 6 Arturs Backurs, Piotr Indyk, Krzysztof Onak, Baruch Schieber, Ali Vakilian, and Tal Wagner. Scalable fair clustering. In *International conference on machine learning*, pages 405–413. PMLR, 2019. URL: <http://proceedings.mlr.press/v97/backurs19a.html>.
- 7 MohammadHossein Bateni, MohammadTaghi Hajiaghayi, and Vahid Liaghat. Improved approximation algorithms for (budgeted) node-weighted steiner problems. *SIAM Journal on Computing*, 47(4):1513–1543, 2018.
- 8 Punam Bedi and Chhavi Sharma. Community detection in social networks. *Wiley interdisciplinary reviews: Data mining and knowledge discovery*, 6(3):115–135, 2016. doi:10.1002/WIDM.1178.
- 9 Suman Bera, Deeparnab Chakrabarty, Nicolas Flores, and Maryam Negahbani. Fair algorithms for clustering. In *Advances in Neural Information Processing Systems*, volume 32, 2019.
- 10 Ioana O. Bercea, Martin Groß, Samir Khuller, Aounon Kumar, Clemens Rösner, Daniel R. Schmidt, and Melanie Schmidt. On the Cost of Essentially Fair Clusterings. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*, volume 145 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 18:1–18:22. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.APPROX-RANDOM.2019.18.
- 11 Diptarka Chakraborty, Kushagra Chatterjee, Debarati Das, and Tien-Long Nguyen. Generalizing fair clustering to multiple groups: Algorithms and applications. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 19934–19942, 2026. doi:10.1609/AAAI.V40I24.39077.
- 12 Diptarka Chakraborty, Kushagra Chatterjee, Debarati Das, and Tien-Long Nguyen. A generic framework for fair consensus clustering in streams. *arXiv preprint arXiv:2602.11500*, 2026. doi:10.48550/arXiv.2602.11500.
- 13 Diptarka Chakraborty, Kushagra Chatterjee, Debarati Das, Tien Long Nguyen, and Romina Nobahari. Towards fair representation: Clustering and consensus. *arXiv preprint arXiv:2506.08673*, 2025. doi:10.48550/arXiv.2506.08673.
- 14 Moses Charikar and Sudipto Guha. Improved combinatorial algorithms for the facility location and k-median problems. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pages 378–388. IEEE, 1999. doi:10.1109/SFFCS.1999.814609.
- 15 Moses Charikar, Sudipto Guha, Éva Tardos, and David B Shmoys. A constant-factor approximation algorithm for the k-median problem. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 1–10, 1999.
- 16 Moses Charikar, Samir Khuller, David M Mount, and Giri Narasimhan. Algorithms for facility location problems with outliers. In *SODA*, volume 1, pages 642–651, 2001. URL: <http://dl.acm.org/citation.cfm?id=365411.365555>.
- 17 Chandra Chekuri, Alina Ene, and Ali Vakilian. Node-weighted network design in planar and minor-closed families of graphs. *ACM Transactions on Algorithms (TALG)*, 17(2):1–25, 2021. doi:10.1145/3447959.
- 18 Fanglan Chen, Subhodip Biswas, Zhiqian Chen, Shuo Lei, Naren Ramakrishnan, and Chang-Tien Lu. Exploring tradeoffs in automated school redistricting: computational and ethical perspectives. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 15912–15920, 2023. doi:10.1609/AAAI.V37I13.26889.

- 19 Ke Chen. On k -median clustering in high dimensions. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 1177–1185, 2006. URL: <http://dl.acm.org/citation.cfm?id=1109557.1109687>.
- 20 Xingyu Chen, Brandon Fain, Liang Lyu, and Kamesh Munagala. Proportionally fair clustering. In *International conference on machine learning*, pages 1032–1041. PMLR, 2019. URL: <http://proceedings.mlr.press/v97/chen19d.html>.
- 21 Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. *Advances in neural information processing systems*, 30, 2017.
- 22 Vincent Cohen-Addad, Fabrizio Grandoni, Euiwoong Lee, Chris Schwiegelshohn, and Ola Svensson. A $(2 + \epsilon)$ -approximation algorithm for metric k -median. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 615–624, 2025. doi:10.1145/3717823.3718299.
- 23 Vincent Cohen-Addad, Anupam Gupta, Lunjia Hu, Hoon Oh, and David Saulpic. An improved local search algorithm for k -median. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1556–1612. SIAM, 2022. doi:10.1137/1.9781611977073.65.
- 24 Zhen Dai, Yury Makarychev, and Ali Vakilian. Fair representation clustering with several protected classes. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 814–823, 2022. doi:10.1145/3531146.3533146.
- 25 Erik D Demaine, MohammadTaghi Hajiaghayi, and Philip N Klein. Node-weighted steiner tree and group steiner tree in planar graphs. In *International Colloquium on Automata, Languages, and Programming*, pages 328–340. Springer, 2009. doi:10.1007/978-3-642-02927-1_28.
- 26 Lukas Drexler, Jan Eube, Kelin Luo, Dorian Reineccius, Heiko Röglin, Melanie Schmidt, and Julian Wargalla. Connected k -center and k -diameter clustering. *Algorithmica*, 86(11):3425–3464, 2024. doi:10.1007/S00453-024-01266-9.
- 27 Seyed Esmaeili, Brian Brubach, Aravind Srinivasan, and John Dickerson. Fair clustering under a bounded cost. *Advances in Neural Information Processing Systems*, 34:14345–14357, 2021. URL: <https://proceedings.neurips.cc/paper/2021/hash/781877bda0783aac5f1cf765c128b437-Abstract.html>.
- 28 Seyed Esmaeili, Brian Brubach, Leonidas Tsepenekas, and John Dickerson. Probabilistic fair clustering. *Advances in Neural Information Processing Systems*, 33:12743–12755, 2020.
- 29 Jan Eube, Kelin Luo, Dorian Reineccius, Heiko Röglin, and Melanie Schmidt. Connected k -median with disjoint and non-disjoint clusters. In *Proceedings of the 33rd Annual European Symposium on Algorithms (ESA)*, LIPIcs, 2025. doi:10.4230/LIPIcs.ESA.2025.63.
- 30 Hossein Fani and Ebrahim Bagheri. Community detection in social networks. *Encyclopedia with semantic computing and robotic intelligence*, 1(01):1630001, 2017.
- 31 Rong Ge, Martin Ester, Byron J Gao, Zengjian Hu, Binay Bhattacharya, and Boaz Ben-Moshe. Joint cluster analysis of attribute data and relationship data: The connected k -center problem, algorithms and applications. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2(2):1–35, 2008. doi:10.1145/1376815.1376816.
- 32 Mehrdad Ghadiri, Samira Samadi, and Santosh Vempala. Socially fair k -means clustering. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 438–448, 2021. doi:10.1145/3442188.3445906.
- 33 Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical computer science*, 38:293–306, 1985. doi:10.1016/0304-3975(85)90224-5.
- 34 Neelima Gupta, Aditya Pancholi, and Yogish Sabharwal. Clustering with internal connectedness. In *International Workshop on Algorithms and Computation*, pages 158–169. Springer, 2011. doi:10.1007/978-3-642-19094-0_17.
- 35 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. doi:10.1287/MOOR.10.2.180.
- 36 Wen-Lian Hsu and George L Nemhauser. Easy and hard bottleneck location problems. *Discrete Applied Mathematics*, 1(3):209–215, 1979. doi:10.1016/0166-218X(79)90044-1.

- 37 Kamal Jain, Mohammad Mahdian, and Amin Saberi. A new greedy approach for facility location problems. In *Proceedings of the thirty-fourth annual ACM symposium on Theory of computing*, pages 731–740, 2002. doi:10.1145/509907.510012.
- 38 Christopher Jung, Sampath Kannan, and Neil Lutz. Service in your neighborhood: Fairness in center location. *Foundations of Responsible Computing (FORC)*, 2020.
- 39 Samir Khuller and Yoram J Sussmann. The capacitated k -center problem. *SIAM Journal on Discrete Mathematics*, 13(3):403–418, 2000. doi:10.1137/S0895480197329776.
- 40 Shi Li. Approximating capacitated k -median with k open facilities. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pages 786–796. SIAM, 2016.
- 41 Sepideh Mahabadi and Ali Vakilian. Individual fairness for k -clustering. In *International conference on machine learning*, pages 6586–6596. PMLR, 2020. URL: <http://proceedings.mlr.press/v119/mahabadi20a.html>.
- 42 Yury Makarychev and Ali Vakilian. Approximation algorithms for socially fair clustering. In *Conference on Learning Theory*, pages 3246–3264. PMLR, 2021. URL: <http://proceedings.mlr.press/v134/makarychev21a.html>.
- 43 S. A. Plotkin, D. B. Shmoys, and É. Tardos. Fast approximation algorithms for fractional packing and covering problems. In *Proceedings of the 32nd Annual Symposium on Foundations of Computer Science (FOCS)*, 1991. doi:10.1109/SFCS.1991.185411.
- 44 Eklavya Sharma. An approximation algorithm for covering linear programs and its application to bin-packing. *arXiv preprint arXiv:2011.11268*, 2020. arXiv:2011.11268.
- 45 Ali Vakilian and Mustafa Yalciner. Improved approximation algorithms for individually fair clustering. In *International conference on artificial intelligence and statistics*, pages 8758–8779. PMLR, 2022.
- 46 Hamidreza Validi, Austin Buchanan, and Eugene Lykhovyd. Imposing contiguity constraints in political districting models. *Operations Research*, 70(2):867–892, 2022. doi:10.1287/OPRE.2021.2141.
- 47 Neal E Young. Sequential and parallel algorithms for mixed packing and covering. In *Proceedings 42nd IEEE symposium on foundations of computer science*, pages 538–546. IEEE, 2001. doi:10.1109/SFCS.2001.959930.

A Proof of Lemma 5

Proof. Fix a feasible overlapping solution $(S_c)_{c \in C}$. By feasibility, for every $c \in C$, we have $c \in S_c$ and $G[S_c]$ is connected. Hence $S_c \in \mathcal{T}_c$ for every center c . Define a configuration-LP solution y by

$$y_{T,c} := \begin{cases} 1, & \text{if } T = S_c, \\ 0, & \text{otherwise.} \end{cases}$$

We verify feasibility. First, for every $v \in V \setminus C$, the overlapping solution satisfies $\bigcup_{c \in C} S_c = V$, so there exists at least one center c such that $v \in S_c$. Therefore, $\sum_{c \in C} \sum_{\substack{T \in \mathcal{T}_c \\ v \in T}} y_{T,c} \geq 1$, and thus the covering constraints (2) hold for all $v \in V \setminus C$. Second, by construction, all variables satisfy $y_{T,c} \geq 0$, so (3) also holds. Finally, the objective value of this LP solution is

$$\sum_{c \in C} \sum_{T \in \mathcal{T}_c} \text{cost}_c(T) y_{T,c} = \sum_{c \in C} \text{cost}_c(S_c) = \sum_{c \in C} \sum_{v \in S_c} d(v, c),$$

which is exactly the cost of the given integral solution.

Since every feasible solution to the assignment variant of connected k -median with overlapping clusters induces a feasible solution to the configuration LP with the same cost, the optimal value of the configuration LP is at most OPT, the optimal cost of connected k -median with overlapping clusters for the prescribed center set C . ◀

B

 Proof of Lemma 8

Proof. Write $w(T) := \sum_{v \in T} w(v)$ and $\alpha(T) := \sum_{v \in T} \alpha_v$. Let T^* be an optimal density solution with $\lambda^* := w(T^*)/\alpha(T^*)$.

We use the bicriteria guarantee of Bateni et al. [7, Theorem 3] for the *rooted budgeted node-weighted Steiner tree problem*. In this problem, we are given a graph $G = (V, E)$, a designated root $c \in V$, a non-negative node-cost function $w : V \rightarrow \mathbb{R}_{\geq 0}$, a non-negative node-profit function $\alpha : V \rightarrow \mathbb{R}_{\geq 0}$, and a budget $B \geq 0$. The goal is to find a connected tree $T \subseteq G$ containing c whose total node cost is at most B and whose total profit $\alpha(T) := \sum_{v \in T} \alpha(v)$ is as large as possible. Let

$$\alpha^*(B) := \max\{\alpha(T) : T \ni c \text{ is connected and } w(T) \leq B\} \text{ where } w(T) := \sum_{v \in T} w(v).$$

Then, for any budget B and any $\varepsilon > 0$, there is a polynomial-time algorithm that returns a connected tree $T_B \ni c$ such that $w(T_B) \leq (1 + \varepsilon)B$ and $\alpha(T_B) \geq \frac{\varepsilon}{O(\log n)} \cdot \alpha^*(B)$. In other words, the algorithm may violate the budget by a factor of at most $1 + \varepsilon$, while guaranteeing an $O(\log n/\varepsilon)$ -approximation to the maximum achievable profit under budget B .

Set $W_{\max} := w(V)$ and try the $O(\log W_{\max})$ budgets $B \in \{1, 2, 4, \dots, 2^{\lceil \log_2 W_{\max} \rceil}\}$. There exists B^* in this list with $w(T^*) \leq B^* \leq 2w(T^*)$. Apply the guarantee of Bateni et al. [7] with $\varepsilon := 1$. Since $w(T^*) \leq B^*$, we have $\alpha^*(B^*) \geq \alpha(T^*)$. The algorithm returns $T_{B^*} \ni c$ with $w(T_{B^*}) \leq 2B^* \leq 4w(T^*)$ and $\alpha(T_{B^*}) \geq \alpha(T^*)/O(\log n)$. Therefore,

$$\text{dens}(T_{B^*}) = \frac{w(T_{B^*})}{\alpha(T_{B^*})} \leq \frac{4w(T^*)}{\alpha(T^*)/O(\log n)} = O(\log n) \cdot \lambda^*.$$

Returning $\arg \min_B \text{dens}(T_B)$ over all tried budgets yields an $O(\log n)$ -approximate density oracle in polynomial time. ◀

C

 Proof of Theorem 10

Proof. The covering LP $\min\{c^\top y : Ay \geq \mathbf{1}, y \geq 0\}$ has (in our usage) exponentially many columns, so we cannot solve it directly; we only have oracle access to an approximately-best column for any given constraint weighting. The proof has three parts.

1. We first observe that the α -approximate *pricing* oracle we are given is indeed also an η -weak *index-finding* oracle ($\eta = 1/\alpha$) for a natural quantity $D_j(w)$, the “coverage per unit weight” of column j .
2. We then show that an index-finding oracle over columns is the same thing as a point-finding oracle over a certain simplex-like polytope P_r (all vectors of cost exactly r). This is the step that lets us invoke a known black-box algorithm for fractional covering, which expects a point-finding oracle, not a column oracle.
3. The black-box algorithm, for a *fixed* guess r of the optimal cost, either certifies cost r suffices (up to a small violation) or fails. We do not know OPT in advance, so we do a binary search over r ; two monotonicity facts about this success/failure signal pin down $r \approx \text{OPT}$ and let us convert the near-feasible solution found at the end into a genuinely feasible one, at a controlled extra cost.

In what follows, we elaborate on each of these steps.

Part 1: from pricing to index-finding. For a column j and weight vector $w \in \mathbb{R}_{\geq 0}^m$, let $D_j(w) := \frac{w^\top A_{\cdot j}}{c_j} = \frac{\sum_i A_{ij} w_i}{c_j}$. Think of $D_j(w)$ as how much weighted coverage column j buys you per unit of its cost. An oracle is an η -weak index-finding oracle if, given $w \geq 0$, it returns some j with $D_j(w) \geq \eta \max_\ell D_\ell(w)$, in other words a column that is within a factor $\eta \leq 1$ of the best possible coverage-per-cost.

The assumed pricing oracle minimizes the *reciprocal* quantity $c_j/(w^\top A_{\cdot j}) = 1/D_j(w)$ to within a factor α : If $\frac{c_j}{w^\top A_{\cdot j}} \leq \alpha \cdot \min_{j'} \frac{c_{j'}}{w^\top A_{\cdot j'}}$, then taking reciprocals (all quantities are positive since $c > 0$; the degenerate case $\max_\ell D_\ell(w) = 0$, i.e. when w is orthogonal to every column, makes the index-finding requirement vacuous and is set aside), then

$$D_j(w) \geq \frac{1}{\alpha} \max_\ell D_\ell(w).$$

So the pricing oracle is an η -weak index-finding oracle with $\eta := 1/\alpha$.

Part 2: from index-finding to point-finding on P_r . Fix a target cost $r \geq 0$ and consider the cost- r slice of the nonnegative orthant, $P_r := \{y \in \mathbb{R}_{\geq 0}^N : c^\top y = r\}$. We introduce this set because known fractional-covering algorithms are stated abstractly for a polytope P together with a *point-finding* oracle (return an approximately-best point of P for a given linear objective), and we want to feed our column oracle into one of them. The set P_r is exactly the right polytope: every $y \in P_r$ can be written as $y = \sum_{j=1}^N \frac{c_j y_j}{r} \cdot \frac{r}{c_j} e_j$, and the coefficients $c_j y_j / r$ are nonnegative and sum to 1 (since $\sum_j c_j y_j = c^\top y = r$), so y is a convex combination of the points $\{(r/c_j) e_j : j \in [N]\}$. In other words, P_r is the simplex whose vertices are, for each column j , spend the entire budget r on column j alone. Maximizing a linear function over a simplex is achieved at a vertex, so for any weight vector $w \geq 0$,

$$\max_{y \in P_r} w^\top A y = \max_{j \in [N]} w^\top A \left(\frac{r}{c_j} e_j \right) = r \cdot \max_j D_j(w).$$

This identity is the crux of this part: finding a good *point* of P_r for objective $w^\top A(\cdot)$ is literally the same problem as finding a good *column* for $D_j(w)$, just rescaled by r . So our η -weak index-finding oracle, applied to w , returns a column j with $D_j(w) \geq \eta \max_\ell D_\ell(w)$; setting $y := (r/c_j) e_j \in P_r$ gives

$$w^\top A y = r D_j(w) \geq \eta \cdot r \max_\ell D_\ell(w) = \eta \max_{y' \in P_r} w^\top A y',$$

In other words, y is an η -weak maximizer of $w^\top A y$ over P_r . We have therefore converted our column oracle into an η -weak point-finding oracle for P_r , for every r simultaneously (the construction above does not depend on r except through the final rescaling).

We also record the *width* of this setup, since the black-box algorithm's running time depends on it: by the same vertex computation,

$$\text{width}(A, \mathbf{1}, P_r) := \max_{y \in P_r} \max_i (A y)_i = \max_{i,j} A_{ij} \cdot \frac{r}{c_j} = r \max_{i,j} \frac{A_{ij}}{c_j}.$$

Using any known bound $q \geq \text{OPT}$ and the given $W \geq q \max_{i,j} A_{ij}/c_j$, we get, for every $r \leq q$,

$$\text{width}(A, \mathbf{1}, P_r) = r \max_{i,j} \frac{A_{ij}}{c_j} \leq q \max_{i,j} \frac{A_{ij}}{c_j} \leq W.$$

Part 3: black-box covering, then binary search on r . We now invoke, as a black box, the standard guarantee for fractional covering with an approximate point-finding oracle (Plotkin–Shmoys–Tardos; we use the clean restatement of Sharma [44], Theorems 3–4): given the η -weak point-finding oracle for P_r from Part 2 and the width bound W from Part 2, and setting $\mu := \frac{\eta}{1+\varepsilon}$, the algorithm either correctly reports $\{y \in P_r : Ay \geq \mathbf{1}\}$ is empty or returns some $y \in P_r$ with $Ay \geq \mu \mathbf{1}$ (i.e. every covering constraint is satisfied up to a $(1 - \mu)$ shortfall) using a number of oracle calls that yields support size $|\text{supp}(y)| = \tilde{O}(mW/(\eta\varepsilon^3))$ when the oracle always returns a single vertex (which ours does, by construction in Part 2). Call this routine on input r and write $g(r) = 1$ if it succeeds (returns such a y) and $g(r) = 0$ if it reports infeasibility.

The routine above only tells us, for a *given* r , whether spending exactly r can nearly cover everything. We do not know OPT, so we search for the smallest r at which $g(r)$ flips from 0 to 1. The bisection argument has two parts:

(i) $r \geq \text{OPT} \implies g(r) = 1$. Let y^* be an optimal solution of the original covering LP, so $c^\top y^* = \text{OPT}$ and $Ay^* \geq \mathbf{1}$. Rescaling, $\frac{r}{\text{OPT}}y^* \in P_r$ (it has cost exactly r) and $A(\frac{r}{\text{OPT}}y^*) = \frac{r}{\text{OPT}}Ay^* \geq \frac{r}{\text{OPT}}\mathbf{1} \geq \mathbf{1}$ since $r \geq \text{OPT}$. So $\{y \in P_r : Ay \geq \mathbf{1}\}$ is genuinely nonempty, and since it is nonempty (and in particular $Ay \geq \mathbf{1} \geq \mu \mathbf{1}$ trivially, as $\mu \leq 1$), the routine cannot correctly report infeasibility; hence $g(r) = 1$.

(ii) $r < \mu \cdot \text{OPT} \implies g(r) = 0$. Suppose instead $g(r) = 1$, i.e. the routine returns $y \in P_r$ with $Ay \geq \mu \mathbf{1}$. Rescale: $\hat{y} := y/\mu$ satisfies $A\hat{y} = Ay/\mu \geq \mathbf{1}$, so $\hat{y}$ is feasible for the *original* covering LP, with cost $c^\top \hat{y} = (c^\top y)/\mu = r/\mu < \text{OPT}$ (using $r < \mu \cdot \text{OPT}$). This contradicts optimality of OPT. So $g(r) = 1$ is impossible, i.e. $g(r) = 0$.

Then, we initialize $\ell = 0$, $u = q$ (both satisfy $g(\ell) = 0, g(u) = 1$, using fact (ii) at $\ell = 0$ whenever $\text{OPT} > 0$), and repeatedly query the midpoint $r = (\ell + u)/2$: if $g(r) = 0$ set $\ell := r$, otherwise set $u := r$ and record the returned vector. Stop once $u \leq (1 + \delta)\ell$ for $\delta := \varepsilon^2/(1 + \varepsilon)$; this takes $O(\log(q/(\delta \cdot (\text{smallest gap}))))$ iterations, polynomial in the input size.

Finally, at termination $g(\ell) = 0$ forces $\ell < \text{OPT}$ (else fact (i) would give $g(\ell) = 1$), so $u \leq (1 + \delta)\ell < (1 + \delta)\text{OPT}$. Let $y^{(u)} \in P_u$ be the vector recorded at the last successful call ($g(u) = 1$), so $Ay^{(u)} \geq \mu \mathbf{1}$ and $c^\top y^{(u)} = u$. Define $\hat{y} := \frac{1}{\mu}y^{(u)}$. Exactly as in fact (ii)'s rescaling, $A\hat{y} \geq \mathbf{1}$, so $\hat{y}$ is feasible for the original covering LP. Its cost is $c^\top \hat{y} = \frac{u}{\mu} \leq \frac{(1+\delta)}{\mu}\text{OPT}$. Substituting $\mu = \eta/(1 + \varepsilon)$ and $\delta = \varepsilon^2/(1 + \varepsilon)$:

$$\frac{1 + \delta}{\mu} = \left(1 + \frac{\varepsilon^2}{1 + \varepsilon}\right) \cdot \frac{1 + \varepsilon}{\eta} = \frac{1 + \varepsilon + \varepsilon^2}{\eta} = \alpha(1 + \varepsilon + \varepsilon^2),$$

using $\eta = 1/\alpha$ from Part 1. This gives the claimed cost bound $c^\top \hat{y} \leq \alpha(1 + \varepsilon + \varepsilon^2)\text{OPT}$. Rescaling by $1/\mu$ does not change the support, so

$$|\text{support}(\hat{y})| = |\text{support}(y^{(u)})| = \tilde{O}\left(\frac{mW}{\eta\varepsilon^3}\right) = \tilde{O}\left(\frac{mW\alpha}{\varepsilon^3}\right),$$

matching the statement of the theorem.

Runtime. Each bisection step invokes the black-box covering routine once, which in turn calls our point-finding oracle (equivalently, the original pricing oracle, once per iteration of that routine) $\text{poly}(m, W, 1/\varepsilon)$ times; the number of bisection steps is $\text{poly}(\log(q/\text{OPT}), \log(1/\delta))$. All oracle calls and arithmetic operations are polynomial whenever q , W , and the input encoding lengths are polynomially bounded, giving the stated overall polynomial running time. $\blacktriangleleft$