Hardness and Approximation Algorithms for Balanced Districting Problems

Prathamesh Dharangutte
Department of Computer Science, Rutgers University, New Brunswick, NJ, USA

Jie Gao
Department of Computer Science, Rutgers University, New Brunswick, NJ, USA

Shang-En Huang
Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan

Fang-Yi Yu
Department of Computer Science, George Mason University, Fairfax, VA, USA

Abstract

We introduce and study the problem of balanced districting, where given an undirected graph with vertices carrying two types of weights (different population, resource types, etc) the goal is to maximize the total weights covered in vertex disjoint districts such that each district is a star or (in general) a connected induced subgraph with the two weights to be balanced. This problem is strongly motivated by political redistricting, where contiguity, population balance, and compactness are essential. We provide hardness and approximation algorithms for this problem. In particular, we show NP-hardness for an approximation better than $n^{1/2-\delta}$ for any constant $\delta > 0$ in general graphs even when the districts are star graphs, as well as NP-hardness on complete graphs, tree graphs, planar graphs and other restricted settings. On the other hand, we develop an algorithm for balanced star districting that gives an $O(\sqrt{n})$-approximation on any graph (which is basically tight considering matching hardness of approximation results), an $O(\log n)$ approximation on planar graphs with extensions to minor-free graphs. Our algorithm uses a modified Whack-a-Mole algorithm [Bhattacharya, Kiss, and Saranurak, SODA 2023] to find a sparse solution of a fractional packing linear program (despite exponentially many variables) which requires a new design of a separation oracle specific for our balanced districting problem. To turn the fractional solution to a feasible integer solution, we adopt the randomized rounding algorithm by [Chan and Har-Peled, SoCG 2009]. To get a good approximation ratio of the rounding procedure, a crucial element in the analysis is the *balanced scattering separators* for planar graphs and minor-free graphs – separators that can be partitioned into a small number of k-hop independent sets for some constant k – which may find independent interest in solving other packing style problems. Further, our algorithm is versatile – *the very same algorithm* can be analyzed in different ways on various graph classes, which leads to class-dependent approximation ratios. We also provide a FPTAS algorithm for complete graphs and tree graphs, as well as greedy algorithms and approximation ratios when the district cardinality is bounded, the graph has bounded degree or the weights are binary. We refer the readers to the full version of the paper for complete set of results and proofs.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis; Theory of computation → Graph algorithms analysis

Keywords and phrases Approximation algorithms, algorithmic fairness

Digital Object Identifier 10.4230/LIPIcs.FORC.2025.4

Related Version *Full Version*: `https://arxiv.org/abs/2501.17277`

Funding *Prathamesh Dharangutte*: Supported by NSF IIS-2229876, DMS-2220271, DMS-2311064, CCF-2208663, CCF-2118953.
Jie Gao: Supported by NSF IIS-2229876, DMS-2220271, DMS-2311064, CCF-2208663, CCF-2118953.
Shang-En Huang: Author conducted part of this research in Boston College and in visiting Osaka University and is currently supported by NTU Grant 113L7491.

1 Introduction

In this paper we study the problem of balanced districting, where we are given an undirected graph where vertices carry two types of weights (population, resource types, etc) and the goal is to find vertex disjoint districts such that each district is a connected induced subgraph with the total weights to be c-balanced – each type of weight is at least $1/c$ of total weight of the district. We aim to maximize the total weights of the vertex disjoint balanced districts.

This problem is an abstraction of many real world scenarios of districting where contiguity/connectivity, population/resource type balance and compactness are desirable properties. For example, in political redistricting, several towns are grouped into a state legislative district or a congressional district. Balancedness requires that each district maintains a sufficient fraction of each (political or demographic) group, which is essential for several reasons. First, voter turnout rates sharply increase if the anticipated election outcome is expected to be a close tie. [14] Thus a balanced district would motivate and raise voter turnout rates. Additionally, balancedness ensures that each political group has the opportunity to elect a candidate of their choice, in compliance with the Voting Rights Act of 1965 [41] and other amendments [25]. This principle also helps to prevent the tipping point in racial segregation, where residents of one demographic group start to leave a district once their population falls below a certain threshold. [56, 52] Connectivity or contiguity, on the other hand, demands that each district be geographically contiguous, – in the language of graph theory that the vertices corresponding to the towns form an induced connected subgraph. This requirement is enforced by most state laws and is a standard practice in general. Many states also have a compactness rule [2], which refers to the principle that the constituents residing within an electoral district should live as near to one another as possible. It often manifests into a preference for regular geometric shapes or high roundness (small ratio of circumference and total area).

The problem of redistricting also appears in many other scenarios such as districting for public schools, sales and services, healthcare, police and emergency services [13], and logistics operations [46]. In addition, the balanced districting problem is of interest in a broader range of applications for resource allocation. For example modern computing infrastructure such as cloud computing provides services to a dynamic set of customers with diverse demands. Customer applications may have a variety of requirements on the combination of different resources (such as CPU cycles, memory, storage, or access of special hardware) that can be summarized by the balanced requirement.

Due to the importance of the problem, redistricting has been studied in a computational sense for schools and elections, that dates back to the 1960s. [42] Since then, an extensive line of work (see [8] for a survey) has formulated the redistricting task as an optimization problem with a certain set of objectives. A lot of existing work considers the geographical map as input and comes up with practical methods and software implementations that generate feasible districting plans. We will survey such work in Section 1.2. However, most previous works focus on optimizing a single desirable property alone (e.g., connectivity [1], or balance [38]), or optimizing average aggregated scores combining multiple objectives of the districts [31]. In contrast, our problem formulation takes these objectives as hard constraints and optimizes the total population that satisfies them. There are several merits of this formulation. First, it offers interpretable, fair, worst-case guarantees for the identified districts. Districting problem is a multi-faceted one. With multiple criteria taken into consideration, it feels ill-fit if only one criterion is singled out as the optimization objective. Furthermore, an average quality guarantee does not provide meaningful utility at the individual district level, and aggregated scores offer limited insight into each objective. Second, a districting solution has a consequential nature and should be taken with a dynamic and time-evolving perspective. Once a new districting plan is in place, residents naturally respond to the algorithm output, resulting in changes in the population distributions. One prominent example is the tipping point theory in racial segregation mentioned above. Optimizing a single balancedness score can still lead to many districts falling below such a tipping point, exacerbating segregation. With this in consideration it is important to keep balancedness as a hard constraint, which hopefully facilitates district stability and integration. With c-balanced property as a requirement, a graph partitioning into vertex disjoint c-balanced districts is not always possible. For example, if the total weight is not c-balanced, some districts have to be unbalanced no matter how the districts are defined. Therefore, we aim to maximize the total weights in balanced districts.

In this paper we focus on the graph theoretic perspective of the redistricting problem. We abstract the input as a graph where vertices represent natural geographical entities/blocks (e.g., townships) and edges of two vertices represent geographical adjacency/contiguity. We focus on two important quality considerations namely connectivity and balanceness, and we maximize coverage, i.e., the total weight (population) covered by balanced districts. In addition, we also consider compactness, which in our setting leads to preference of districts as low-diameter subgraphs. An important case studied in this paper is to consider a balanced *star district*, which consists of a center vertex v as well as a set of neighboring blocks all adjacent to v. We also consider districts of bounded rank k for a constant k – where a district has at most k vertices.

1.1 Our Results and Technical Overview

We report a systematic study of the balanced districting problem on both hardness results and approximation algorithms. Our goal is to dissect the problem along different types of input graph topologies (general graphs, planar graphs, bounded degree graphs, complete graphs, tree graphs, etc), district types (e.g., arbitrarily connected districts, star districts, or bounded rank-k), and weight assumptions (arbitrary weights, binary weights). A brief summary of our results can be found in Table 1.

Complexity and Challenges

There are three elements in the balanced districting problem that make it challenging and interesting, from a technical perspective: 1) connectivity – the induced subgraph of a district is connected 2) packing and coverage maximization – no vertex belongs to two districts and we maximize the total weights of included vertices; 3) balancedness – the two types of weights in a district need to be roughly balanced. These elements are shared with a number of well known hard problems, suggesting that our problem is also computationally challenging. For example, the exact set cover problem asks if there is a perfect coverage and packing in a set cover instance. The packing element is shared with maximum independent set problem – a vertex included will forbid its neighbors to be included. And the balancedness is shared with subset sum problem. Therefore by using the hardness of these problems we can show hardness and hardness of approximation of the balanced districting problem for a variety of graph classes. The hardness of the balanced districting problem is immediately shown by a reduction from exact set cover problem. By a reduction from maximum independent set problem, we can show that the balanced districting problem does not have an approximation of $n^{1/2-\delta}$ for any constant $\delta > 0$ in a general graph of n vertices unless $\mathsf{P} = \mathsf{NP}$ and is

Table 1 A summary of hardness and approximation results on the balanced districting problem. $\delta, \varepsilon > 0$ are constants. Tight results are highlighted in bold. We only provide a subset of these results in this version, please refer to the full version for the complete set of results.

Graph Type	District Type	Result
General	arbitrary/star	**NP-hard for $n^{1/2-\delta}$-approx**
Max degree Δ	arbitrary/star	APX-hard for $\Delta = O(1)$ NP-hard for $\Delta/2^{O(\sqrt{\Delta})}$-approx UGC-hard for $O(\Delta/\log^2\Delta)$-approx
Planar with $\Delta = 3$	star with rank-3	NP-hard
Complete or Tree	arbitrary/star	**NP-hard**
Planar	star	$O(\log n)$-approx
H-Minor-Free	star	$O(h^2 \log n)$-approx, $h = \|H\|$
Outerplanar	star	$O(1)$-approx
General	star	**$\Theta(\sqrt{n})$-approx**
Complete graph	arbitrary/star	**FPTAS $(1+\varepsilon)$-approx**
Tree	arbitrary/star	**FPTAS $(1+\varepsilon)$-approx**
General	rank-2	polynomially solvable
General	rank-k, $k > 2$	k-approx
Bounded degree Δ	star	$(\Delta + \frac{1}{\Delta})$-approx
General (binary weights)	star	c-approx

APX-hard for bounded degree graphs. From a reduction from the planar 1-in-3SAT problem, the balanced districting problem is NP-hard for a planar graph, and even if each district has at most three vertices – a crucial condition since the problem can be solved exactly by maximum weighted matching if each district is only allowed to have two vertices. If the input graph is a tree or a complete graph – extremely simple topologies for which many NP-hard problems can be solved in polynomial time, the balanced districting problem is still NP-hard due to the balancedness requirement by a reduction from subset sum. All of the hardness proofs hold even if we limit the output districts to be only of a star topology. This set of hardness results can be seen from the Top section of Table 1.

Greedy Methods on Special Cases

On the positive side, it is natural to ask if existing techniques for solving or approximating these related problems can be borrowed for the balanced districting problem. The answer turns out to be often "not really", even if we only look for balanced star districts. The additional requirements in our problem often break some crucial steps. For example, the problem of maximum independent set has an easy $\Omega(n)$ lower bound if the input graph is sparse (or planar) – thus a simple random greedy algorithm with conflict checking gives an easy constant approximation algorithm. But such lower bound no longer holds true for our problem if the weights are not balanced. Even for star districts, when the maximum degree is not bounded by a constant, the number of potential balanced districts can be exponential in n, the size of the network.

We show in Section 6 of the full version that ideas using a greedy approach and local search method give us approximation algorithms, but only for very special cases. Namely, if the districts have rank-k, we can try the greedy maximum hypergraph matching to have a k-approximation to the optimal solution. When all weights are binary (1 or 0), we can use a greedy algorithm with local search to get a c approximation to the optimal c-balanced star districting solution. Similarly, if the graph has maximum degree Δ, we can get a $(\Delta + \frac{1}{\Delta})$-approximation for c-balanced star districting.

LP Framework and Rounding

To really tackle the problem with arbitrary weights, for districts that are not limited by rank and graphs beyond constant bounded degree, we first examine what we can do with complete graphs or tree graphs – here the topology is made some of the simplest possible, and we would like to address the challenge from packing and balancedness. In this setting we can obtain FPTAS for both complete graphs and tree graphs (Section 5)– although the algorithm is much more involved due to the additional requirements of packing and connectivity (for tree graphs, as connectivity is trivial for complete graphs). Our FPTAS uses a dynamic programming technique that maintains one district's possible weights and introduces a new prioritized trimming method to approximate weights while ensuring that the resulting district satisfies the c-balanced constraint and approximates the optimal weight. The FPTAS for the complete graph is later used as a subroutine for solving the relaxed LP formulation for other graph settings.

Beyond complete graphs and tree graphs, we develop a general framework (Section 4) that produce approximation algorithms for star districts on different types of graphs. All these algorithms start from a relaxed linear program where we formulate a variable x_S for each potential balanced star district S, which can take non-integer values and for all districts that share the same vertex, the sum of their variables is at most 1. Despite potentially exponentially many variables (and constraints in the dual program) that preclude standard solutions, we adapt the whack-a-mole framework [9], which can be seen as a lazy multiplicative weight update algorithm, on dual variables, and we design a "separation oracle" that selects a violating constraint in the dual program in time polynomial in n and $1/\varepsilon$ that can significantly improve the solution. Consequentially, the linear program can be solved in time polynomial in n and $1/\varepsilon$ up to any precision $1-\varepsilon$ and the number of non-zero primal variables (i.e., the candidate balanced star districts) is also polynomial. Intriguingly, our separation oracle is based on our FPTAS on compute graphs for balanced districting.

Now we will round the fractional solution to an integer solution and in the process we may lose an approximation factor. We use a simple randomized rounding method where we sort the districts with non-zero values in decreasing order of total weight, and flip a coin with probability proportional to x_S to include a potential district S, if S does not overlap with any districts already included. In order to bound the loss of quality in the rounding process, we need to upper bound the correlation of the variables, namely, sum of $x_A \cdot x_B$ for all pairs of overlapping districts A, B. These are the (fractional) districts that have to be dropped due to conflict. To establish the approximation factor, we wish to bound the total sum of correlation by a factor multiplied with the total sum over all possible districts $\sum_S x_S$ – exactly the optimal LP solution. We show that this ratio is $O(\sqrt{n})$, which immediately gives an $O(\sqrt{n})$-approximate solution for star districts on a general graph. Notice that this is tight due to the hardness of approximation results.

Due to the strong motivation from political redistricting and resource allocation considering geographical proximity/constraints, the planar graph is of particular interest to us. One of the main technical contributions is a polylogarithmic-approximation algorithm for balanced star districting on planar graphs and related algorithms for minor-free graphs and outer planar graphs. For a planar graph we now adopt a balanced planar separator and use a divide-and-conquer analysis. Namely, we only need to analyze the overlapping districts with at least one of them including vertices in the separator. Now a crucial observation is, if we can partition the planar separator into k 5-hop independent sets, then we can decompose the total sum of correlation by the independent sets – fix an 5-hop independent set X, two star districts that touch different vertices in X are disjoint and star districts that share the same vertex in X have their total district value bounded by 1 due to the primal constraint. This allows us to upper bound the correlation term for the star districts touching the separator by a factor of k of the sum of x_S with districts S on the separator. Recursively, this gives an $O(\log n)$ factor loss in the final approximation.

We remark that the above analysis asks for a new property of a balanced separator – one that can be decomposed into a small number of 5-hop independent sets (called a "scattering" separator) – and we do not care about the size of the separator. This is possible for a planar graph if we use the fundamental cycle separator, which is composed of two shortest paths, and thus at most 10 5-hop independent sets. For a H-minor-free graph with H as a graph of h vertices, we show the existence of a similar separator, which can be decomposed into $O(h^2)$ 5-hop independent sets. Thus the final approximation ratio for H-minor-free graphs is $O(h^2 \log n)$. For outer planar graphs, we can skip the recursive step and work with graph partitions with 5-hop independent sets and get $O(1)$-approximation. We believe that this technique of using balanced scattering separators is interesting in its own and may find additional applications in other problems with some packing (non-overlapping) requirement on the solution.

On general graphs, the formulated linear program could have an integrality gap as large as $\Omega(\sqrt{n})$. Since our rounding algorithm turns an fractional solution into an integral one, this barrier unavoidably blends into our analysis to the proposed rounding algorithm, producing an provable $O(\sqrt{n})$ bound. However, by thinking about this argument contrapositively, an upper bound to our rounding algorithm leads to the integrality gap of the formulated LP, which could be an interesting takeaway. On the other hand, we also show that there are planar graphs (specifically grid graphs) such that our rounding algorithm produces a > 1 constant approximation ratio. This observation suggests that we cannot hope for a PTAS using this approach, even on planar graphs.

1.2 Related work

To the best of our knowledge, this paper is the first to study the balanced districting problem. Below, we briefly survey related problems and explore their potential connections to ours.

Districting

Our problem is connected to computational (re)districting for schools and elections, which dates back to the 1960s. [42] Since then, an extensive line of work (see [8] for a survey) has formulated the redistricting task as an optimization problem with a certain objective and constraints, e.g,. balancedness, contingency, or compactness. Our redistricting problem focuses on optimizing the population in balanced and contiguous districts. One concept related to our notion of balance is competitiveness. Recent work introduces vote-band metrics [30], which require a certain fraction of votes to fall within a specified range (e.g., 45-55%) for competitive elections. Subsequently, [22] also adopt similar notions called δ-Vote-Band Competitive which is equivalent to our c-balancedness by setting $c = 2/(1 - 2\delta)$. While related, our work diverges technically, offering both hardness and algorithmic results for several common graphs. [30] empirically evaluates ensemble methods for district distributions. [22] explored the hardness and heuristic algorithms for maximizing the number of districts meeting the target competitiveness constraints, with additional requirements that all districts have roughly the same population limited compactness consideration.

One approach treats contiguity as a transportation cost and designs linear programming models to minimize this total cost [1, 34, 24]. Interestingly, the fair clustering problem can also be viewed as optimizing contingency [12, 19, 44, 18]. Other research focuses on optimizing compactness scores [6, 45, 48, 43] or using Voronoi or power diagrams with some variant of k-means [57, 26, 27, 35]. Finally, another line of work optimizes balance scores [38]. These approaches differ from ours in that they treat specific aspects of districting (contiguity, compactness, balance, etc.) as objectives, rather than maximizing the population that meets these criteria.

Besides the optimization approach, another popular approach uses sampling to generate a distribution over districts and create a collection of district plans for selection. [3, 23] One widely used method is ReCom [31], an MCMC algorithm. However, these approaches may suffer from slow mixing times and lack formal guarantees. [51] Finally, several papers take a fair division approach [49, 53, 29]. The problem is quite different, however, as fairness is defined concerning parties (types) and the number of seats they would win (i.e., the number of districts where they would have a majority) compared to other districts.

Algorithms

Beyond districting problems, as outlined in the technical overview, our problem connects to several classical algorithm problems. If we only want to maximize the population of a single connected and balanced district, the problem becomes a balanced connected subgraph problem [10, 11, 50]. However, the previous work in this area typically considers unit weights for either type, which does not adequately represent the districting problem that operates in an aggregated block-level setting. Our problem can be seen as packing subgraphs on graph [28], e.g., edges (maximum matching), triangles [47], circles [32]. Finally, we note a line of work on a balanced, connected graph partition [17, 21], and balanced bin-packing problem [33], which, however, aim to generate a partition where each component has similar weights.

1.3 Open Problems

As the first work to formally study the balanced districting problem in this formulation, our work leaves a number of interesting open problems for future work. Obviously it is good to close the gap of approximation and hardness for different families of graphs. Our results are tight for general connected districts on complete graphs and tree graphs, as well as star districts on general graphs, but leave gaps for other settings. We conjecture that there exists an algorithm with a constant approximation factor for c-balanced star districting on planar graphs. It would also be interesting to develop algorithms to go beyond star districts, i.e., k-hop graphs for a constant k or the more general setting of connected districts. We remark that the scattering separator can be modified to handle k-hop graphs but we need an efficient separation oracle. We consider two types of weights/populations and generalizing the problem and solutions to three or more colors would be interesting and is currently widely open. We remark that the PTAS algorithm for complete graphs is specific for two weights/colors. Finally, an interesting future direction would be to develop algorithms that also demand approximate population equality among districts.

2 Preliminaries

Let $G = (V, E)$ be an undirected graph where we call the vertices *blocks*. A *district* $T \subseteq V$ is a subset of blocks where the induced subgraph $G[T]$ is connected. If there exists a block $x \in T$ that is a neighbor of every other block in $T \setminus \{x\}$, then we say T is a *star district* and x is a *center* of T. The *rank* of a district T is the number of blocks in T. A *(partial) districting*

$\mathcal{T}$ is a collection of disjoint districts. That is, $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$ where $T_i \cap T_j = \emptyset$ whenever $i \neq j$. Notice that a districting is not necessarily a partitioning of the graph, i.e., not all blocks are included in the districts.

c-Balanced Districting Problems

There are two communities or commodities of interest. Let the functions $p_1, p_2 : V \to \mathbb{Z}_{\geq 0}$ represent the *population of each community* or *weight of each commodity* on each vertex. The *weight* of a block $w(x)$ is defined to be $p_1(x) + p_2(x)$. Let $T \subseteq V$ be a district. By a natural extension we define $p_i(T) := \sum_{x \in T} p_i(x)$ for $i \in \{1, 2\}$ and $w(T) := p_1(T) + p_2(T)$ accordingly as the weight of the district T. Finally, given a districting $\mathcal{T}$, we define $w(\mathcal{T}) := \sum_{T \in \mathcal{T}} w(T)$ to be the total weight.

Given a constant $c \geq 2$, we say that a district T is *c-balanced* if

$$\min\{p_1(T), p_2(T)\} \geq \frac{w(T)}{c} \ . \tag{1}$$

$\mathcal{T}$ is a *c-balanced districting* if all districts $T \in \mathcal{T}$ are c-balanced. Notice that if the total weights in the graph are not c-balanced, we cannot hope to include all blocks in c-balanced districts. Given $c \geq 2$, a graph G, and functions p_1 and p_2, the problem c-BALANCED-DISTRICTING is subjected to find any c-balanced districting $\mathcal{T}$ that *maximizes* $w(\mathcal{T})$. That is, we wish to maximize population covered in the c-balanced districts.

We will investigate several variants to the problem. A lot of our results concern restricting the output districting to be *star shaped*, respecting the need for compactness of the districts. We also consider districts of bounded rank k if every district has at most k vertices with k assumed to be a constant. In general we consider the weights of the vertices to be arbitrary integer values. A special case is when all weights are uniform (binary) – each vertex has only one non-zero weight type, either $p_1(x) = 0$, $p_2(x) = 1$ or $p_1(x) = 1$, $p_2(x) = 0$.

Let X be any variant to the c-balanced districting problem. A districting $\mathcal{T}$ is said to be *feasible* on X if $\mathcal{T}$ satisfies all districting type constraints, but not necessarily to have its weight maximized. Any districting with the maximum possible total weight is said to be *optimal*. We say that a feasible districting $\mathcal{T}$ is an *f-approximated* solution if $f \cdot w(\mathcal{T}) \geq w(\mathcal{T}_{\mathsf{OPT}})$, where $\mathcal{T}_{\mathsf{OPT}}$ is any optimal districting.

Graph Types

A graph $G = (V, E)$ is said to be *planar* if there exists an embedding of all vertices to the Euclidean plane such that all edges can be drawn without intersections other than the endpoints. A *face* of an planar embedding is a connected region separated by the embedded edges. G is said to be *outerplanar* if there exists an embedding of G such that there is a face containing all vertices. Often this face is assumed to be the outer face. A graph H is said to be a *minor* of G if H is isomorphic to the graph obtained by a sequence of vertex deletions, edge contractions, and edge deletions from G. We say that G is *H-minor-free* if G does not have H as its minor.

3 Hardness Results

We first present hardness results for a variety of c-balanced districting problems with increasing restrictions on the parameters. The proof of Theorem 4 is deferred to Section A while rest of the proofs appear in Appendix A of the full version.

▶ **Theorem 1.** *The c-balanced districting problem is* NP*-hard, for both the case when the districts are connected subgraphs and when the districts are required to be stars.*

Theorem 1 uses a reduction from the ExactSetCover problem. The ExactSetCover problem remains NP-hard even when each set has *exactly* three elements and no element appears in more than three sets [37], or when each element appears in exactly three sets [39]. Therefore, if we limit that each district has at most *four* blocks, the problem remains NP-hard. In the following we show that the problem remains hard if each district has at most *three* blocks and the graph is planar. Notice that if each district has at most two blocks, the problem can be solved by maximum matching in polynomial time.

▶ **Theorem 2.** *The c-balanced districting problem is* NP*-hard, when G is a planar graph with maximum degree* 3*, each district has rank-*3 *(i.e., with at most three blocks), and the districts must be stars.*

The proof of the above claim uses reduction from planar 1-in-3SAT. We show next that the problem on a complete graph or a tree remains hard. This reduction uses the problem of subset sum.

▶ **Theorem 3.** *The c-balanced districting problem is* NP*-hard for any $c \geq 2$, when G is a complete graph or a tree. This holds for both the case when the the districts are connected subgraphs and when the districts are required to be stars.*

Last, we show hardness of approximation by a reduction from the maximum independent set problem.

▶ **Theorem 4.** *The c-balanced districting problem does not have an $n^{1/2-\delta}$-approximation (for any constant $\delta > 0$) in a general graph unless* P = NP*. On a graph with maximum degree Δ, one cannot approximate the c-balanced districting problem within a factor of $\Delta/2^{O(\sqrt{\Delta})}$ assuming* P ≠ NP*, and $O(\Delta/\log^2 \Delta)$ assuming the Unique Games Conjecture (UGC). Even if Δ is a constant, the problem is* APX*-hard. These statements hold when the districts must be stars and when the centers of the stars are limited to a subset of vertices.*

4 An Algorithm for c-Balanced Star Districting

In this section, we give an approximation algorithm to the c-balanced star districting problem. The algorithm is based on a multiplicative weights update approach of solving packing-covering linear programs [54, 4, 9] and then apply a randomized rounding procedure [15]. Interestingly, the same algorithm achieves different approximation guarantees on different classes of the graphs, summarized in the following theorem.

▶ **Theorem 5.** *Let G be a graph with weight functions p_1 and p_2. There exists a polynomial time algorithm that computes a c-balanced star districting $\mathcal{T}$, with the following guarantee:*
(1) *For any general graph G, $\mathcal{T}$ is an $O(\sqrt{n})$-approximate solution.*
(2) *If G is planar, then $\mathcal{T}$ is an $O(\log n)$-approximate solution.*
(3) *If G is an H-minor-free graph with $|H| = h$, then $\mathcal{T}$ is an $O(h^2 \log n)$-approximate solution.*
(4) *If G is a tree or an outerplanar graph, then $\mathcal{T}$ is an $O(1)$-approximate solution.*

In Section 4.1, we formulate the problem as a linear program. In Section 4.2 of the full version, we describe how to apply the Whack-a-Mole algorithm [9] (with our own separation oracle) that obtains an $(1 + \varepsilon)$-approximate solution to the linear program in polynomial time. In Section 4.2, we apply Chan and Har-Peled's randomized rounding technique [15], showing that bounding the pairwise product terms leads to the desired approximation factors. We show that there is a bound of $O(\sqrt{n})$ on any graph in Section 4.3, To bound the pairwise product terms, we introduce the scattering separators in Section 4.4. These scattering separators are useful for analyzing the approximation ratios for planar graphs and for minor-free graphs. For the graph classes that is a subclass to the planar graphs, we provide tailored-but-better analysis for outerplanar graphs and for trees , which conclude the proof of Theorem 5. We refer readers to the full version for complete details of these cases.

4.1 LP Formulation

We formulate the c-balanced districting problem as a linear program. For each c-balanced star district S, we define a variable x_S indicating whether or not this district is chosen. Thus, the integer linear program for can be defined as:

$$\begin{aligned} &\text{maximize} \sum_{S} w(S)x_S \\ &\text{subject to } \forall v \in V, \sum_{S \ni v} x_S \leq 1 \\ &\qquad\qquad \forall S, x_S \in \{0,1\} \end{aligned} \tag{2}$$

To give an approximate solution to the above integer linear program, we follow the standard recipe that solves the relaxed linear program first and then apply a randomized rounding algorithm.

Equivalent Relaxed Linear Program

In order to solve the relaxed linear program of Equation (2), we use *weighted* variables: for each district S, we define variable $x'_S := w(S)x_S$. Hence, the equivalent linear program (and its dual linear program) we will be solving can be described as follows.

$$\begin{array}{ll|ll} & \text{(PRIMAL)} & & \text{(DUAL)} \\ \text{maximize} & \sum_{S} x'_S & \text{minimize} & \sum_{v} y'_v \\ \text{subject to} & \forall v \in V, \sum_{S \ni v} \frac{1}{w(S)} x'_S \leq 1 & \text{subject to} & \forall S, \sum_{v \in S} \frac{1}{w(S)} y'_v \geq 1 \\ & x'_S \geq 0 & & y'_v \geq 0 \end{array}$$

We note that the total number of primal variables (i.e., the number of potential c-balanced star districts) could be exponentially many in terms of the graph size. However, due to the special structure of this problem, seeking for an approximate solution does not require the participation of every variable. We summarize the result of solving the relaxed linear program as Theorem 6 below.

▶ **Theorem 6.** *Given a graph G and a precision parameter $\varepsilon \in (0, \min\{\frac{1}{2}, \frac{c-2}{c}\})$, there exists an algorithm that returns an $(1-\varepsilon)$-approximate solution $\{x'_S, y'_v\}$ to the above linear program in* $\text{poly}(n, 1/\varepsilon, \log(w(G)))$ *time. Moreover, there are at most* $\text{poly}(n, 1/\varepsilon)$ *non-zero terms among the returned primal variables $\{x'_S\}$.*

Implementing the Separation Oracle

In order to have a polynomial time algorithm, we want an efficient separation oracle. The following lemma (Lemma 7) reduces the task of finding a violating district to solving the c-balanced star districting problem in a complete graph.

▶ **Lemma 7.** *Given an input instance $G = (V, E, (p_1, p_2))$, re-weighted values $w' : V \to \{0\} \cup [\frac{1}{w(G)}, w(G)]$, and dual variables $y'_v \in [n^{-(1+1/\varepsilon)}, 1 + \varepsilon]$ for each vertex $v \in V$, there exists an algorithm that either reports that $\mathcal{S}_{\text{violate}} = \emptyset$, or returns a c-balanced district S such that $\sum_{v \in S} y'_v < (1 - \varepsilon/2)w'(S)$ and $w'(S) \geq \frac{1}{2}w'(S_{\max})$, where $S_{\max} = \arg\max_{S \in \mathcal{S}_{\text{violate}}} w'(S)$ is a violating c-balanced district with the maximum value. The algorithm runs in $O(\varepsilon^{-6} n^6 (\log n)(\log w(G))^4)$ time.*

4.2 The Randomized Rounding Algorithm

We use a randomized rounding technique modified from [15, Section 4.3]. Intuitively, the algorithm maintains a set I of non-overlapping districts, which is initially empty, and keeps adding districts into I.

The rounding algorithm is described as follows. Let $\{x_S\}$ be the output of an approximate solution to LP. Let $\mathcal{S}_{\text{LP}} = \{S \mid x_S \neq 0\}$ be the support of the solution. The algorithm first sorts all non-zero valued districts according to their weights $w(S)$, from the largest to the smallest. Let $\tau \geq 1$ be a parameter to be decided later. For each district S, with probability x_S/τ, the algorithm adds S into I as long as there is no district in I overlapping with S. [1] The algorithm outputs I after all districts in $\mathcal{S}_{\text{LP}}$ are considered. The necessity of scaling the non-zero variables by τ comes from the analysis of expected total weight in I. In an actual implementation of the algorithm, one can make the algorithm oblivious of τ, by iteratively testing on different values of $\tau = (1 + \varepsilon)^k$ for $k = 0, 1, 2, \ldots, O(\varepsilon^{-1} \log n)$ and then picking the largest weighted districting among the returned ones.

Analysis

The output I of the algorithm can be seen as a random variable. Let $w(I)$ be the total weight of the districts within I. A straightforward analysis (see Lemma 15) shows that

$$\mathbf{E}[w(I)] \geq \sum_{S \in \mathcal{S}_{\text{LP}}} w(S) \frac{x_S}{\tau} - \sum_{A, B \in \mathcal{S}_{\text{LP}}:\ A \cap B \neq \emptyset} \min(w(A), w(B)) \frac{x_A x_B}{\tau^2} .$$

The right hand side of the above expression contains a weighted correlation term. The technique by Chan and Har-Peled [15] transforms the above weighted correlation terms into *unweighted* ones. They mentioned that, a desired $O(\tau)$-approximate solution can be achieved, as long as for any δ-*thresholded* subset $\mathcal{S}_{\geq \delta} := \{S \in \mathcal{S}_{\text{LP}} \mid w(S) \geq \delta\}$, the total unweighted correlation terms between overlapped districts can be upper bounded by the sum over all primal variables within the subset:

$$\sum_{A, B \in \mathcal{S}_{\geq \delta} : A \cap B \neq \emptyset} x_A x_B \leq \frac{\tau}{2} \cdot \sum_{S \in \mathcal{S}_{\geq \delta}} x_S \quad \forall \delta > 0 \tag{3}$$

[1] The randomization step appears to be necessary since there is an example where deterministic rounding incurs a large approximation factor (examples and details in the full version).

The above condition implies the following (see Lemma 16):

$$\mathbf{E}[w(I)] \geq \frac{1}{2\tau} \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) \cdot x_S \geq \frac{1}{2\tau(1+\varepsilon)} \mathrm{OPT}_{\mathrm{LP}},$$

where $\mathrm{OPT}_{\mathrm{LP}}$ is the optimal value of the LP problem. Thus, I is an $O(\tau)$-approximate solution in expectation. We remark that due to the factor 2 appearing in the denominator, this linear program with randomized rounding approach (although not contradicting) is unlikely to achieve a PTAS.

The above analysis to the rounding algorithm enables the approach of seeking a suitable τ value, such that Equation (3) holds. The rest of the section focuses on providing upper bounds of τ on various classes of input graphs.

4.3 An $O(\sqrt{n})$-Approximation Analysis for General Graphs

In this section, we show that the algorithm achieves an $O(\sqrt{n})$-approximate ratio on any graph, by giving an upper bound $\tau = O(\sqrt{n})$ for the randomized rounding algorithm (with proof in the full version).

▶ **Lemma 8.** *Let G be any graph. Let $\{x_S\}$ be any feasible solution to the linear program. Then,*

$$\sum_{A,B:\ A \cap B \neq \emptyset} x_A x_B \leq \sqrt{n} \cdot \sum_S x_S.$$

Remarks: Integrality Gap

We remark that this algorithm is achieving nearly the best approximation factor since it is NP-hard to have approximation factor of $n^{1/2-\delta}$ for any constant $\delta > 0$. Related to this, we would like to examine the potential loss in different steps of our algorithm. In the first step, we relax the integer linear program to a linear program with variables taking real numbers, the *integrality gap* refers to the ratio of the optimal fractional solution to the optimal integer solution (since we consider maximization problem the optimal fractional solution is no smaller than the optimal integer solution). In the second step of our algorithm, we use randomized rounding to turn the fractional solution back to a feasible integer solution. We call the ratio between the sum of products between overlapping districts' primal variables and the sum of all variables to be the *rounding gap*, i.e., $\sum_{A,B:A\cap B\neq\emptyset} x_A x_B / \sum_S x_S$. The rounding gap can be used to upper bound the loss of solution quality when we turn the fractional solution to a feasible integer solution using the randomized rounding algorithm.

Necessarily, a large integrality gap implies a large rounding gap for sure. Specifically, let τ be the rounding gap. The analysis to our rounding algorithm guarantees the existence of an integral solution within a factor of $4(1+\varepsilon)\tau$ from the optimal fractional solution, which implies an integrality gap of at most 4τ when setting $\varepsilon \to 0$. Thus if the integrality gap is large, we cannot have a small rounding gap. Interestingly, the above discussion, combined with Lemma 8 implies that the integrality gap of the natural LP formulation for the star districting problem is at most $O(\sqrt{n})$.

Next we show that our LP relaxation could have a large integrality gap of $\Omega(\sqrt{n})$ for a general graph. We use a reduction from k-uniform hypergraph matching problem to our c-balanced star districting problem. Let $H = (V_H, E_H)$ be the given k-uniform hypergraph – a hypergraph such that all its hyperedges have size k. We construct a graph $G = (V_H \cup E_H, E_G)$ by creating additional vertices for each hyperedge. These vertices have heavy weights, say

$p_2(e) := (c-1)k$ and vertices from V_H have weights $p_1(v) := 1$. For each hypergraph $e \in E_H$ (which is a subset of vertices), we connect all vertices $v \in e$ to the corresponding vertex e in G. It ensures that there is an one-to-one correspondence between hyperedges of H to c-balanced star districts on G. Now, the relaxed linear program for (G, p_1, p_2) will be equivalent (with an extra ck-factor in the objective function) to a fractional hypergraph matching. Thus, the $(k + 1 - 1/k)$ integrality gap of k-uniform hypergraph matching [36, 16] can be transferred to our LP formulation – specifically, the construction in [16] via projective planes leads to an $\Omega(\sqrt{n})$ integrality gap.

On the other hand, we do observe *planar graph instances* with a constant > 1 rounding gap with an at most $1 + o(1)$ integrality gap (refer the full version). Again, this does not eliminate the possibility of achieving PTAS, but it suggests a conjecture that we are unlikely to obtain a PTAS for planar graphs using the current analysis.

4.4 Scattering Separators

Let us now introduce the scattering separators, which is useful for the divide and conquer framework for upper bounding the approximation ratio of the randomized rounding procedure.

▶ **Definition 9.** *Let $G = (V, E)$ be a graph and let $X \subseteq V$ be any subset. We say that X is:*

- (k, t)-scattered*, if X can be partitioned into at most k subsets $X = X_1 \cup X_2 \cup \cdots \cup X_k$ with each X_i being a t-hop independent set*[2]*;*
- (k, t)-orderly-scattered*, if there exists a way to partition X into a sequence of at most k subsets $X = X_1 \cup X_2 \cup \cdots \cup X_k$, where each X_i is a t-hop independent set after the removal of all previous subsets $G - \cup_{j<i} X_j$.*

▶ **Definition 10.** *Let G be a graph, $k, t \in \mathbb{N}$, and $\delta \in (0, 1)$. A (k, t, δ)-*scattering separator *is a subset of vertices $X \subseteq V$ such that (1) X is (k, t)-orderly-scattered, and (2) X is a balanced separator of G, that is, the largest connected component of $G - X$ has at most δn vertices.*

We remark that a (k, t)-scattered set is also (k, t)-orderly-scattered. This orderly-scattered definition are useful when we remove subsets of vertices sequentially – they are used in the analysis of, for example, planar graphs and minor-free graphs. On the other hand, for some graph class such as outerplanar graphs it suffices to use (k, t)-scattered sets within the analysis.

The scattering separators are useful in the c-balanced districting problem for $t \geq 5$. To justify this, suppose that we have a 5-hop independent set Y. Any star district contains at most one vertex in Y. If two star districts contain different vertices of Y, the two districts must be disjoint. Thus we partition the pairs of overlapping districts by whether they overlap with Y or not, and if so, which vertex of Y. The following fact can be easily verified.

▶ **Fact 11.** *Let Y be a 5-hop independent set. Consider a fixed district $A \in \mathcal{S}$. Assume there is a district $B \in \mathcal{S}$ that overlaps with A and $A \cup B$ touches Y, i.e., $A \cap B \neq \emptyset$ and $(A \cup B) \cap Y \neq \emptyset$. Since the diameter of $G[A \cup B]$ is at most 4, we know that $|(A \cup B) \cap Y| = 1$. Further, if A overlaps with two other star districts B, C with both centers of B, C in Y, then B, C have the same center.*

Fix a district $A \in \mathcal{S}$. Since all other districts that overlap with A contains (at most) the same vertex in Y, these primal variable values add up to at most 1. This implies that, removing Y from G charges at most one copy of $\sum x_S$. If we are able to show that the

[2] We say that X is a *t-hop independent set* (with respect to the graph G) if for all pairs of distinct vertices $u, v \in X$ and $u \neq v$, the shortest distance between u and v is at least t on G.

entire vertex set is a $(k, 5)$-orderly-scattered, then we obtain a desired $\tau = O(k)$ value for Equation (3). However, we do not know if such a constant k can be achieved for planar graphs. Fortunately, using the idea of balanced separators, we are able to achieve a polylogarithmic approximate solution.

▶ **Lemma 12.** *If G and all its subgraphs have a $(k, 5, \delta)$-scattering separator, then the districting obtained from executing the algorithm on G is a $(2k \log_{1/\delta} n)$-approximated solution.*

Proof. Let $X = X_1 \cup X_2 \cup \cdots \cup X_k$ be a $(k, 5, \delta)$-scattering separator of G. Let $\mathcal{S}$ be the set of all districts. Then, all summands of the form $x_A x_B$ where $A, B \in \mathcal{S}$ and $A \cap B \neq \emptyset$ can be also split into three parts:

(1) $X \cap \{c_A, c_B\} \neq \emptyset$: one of the centers c_A or c_B is in X.
(2) $X \cap \{c_A, c_B\} = \emptyset$ but $X \cap A \cap B \neq \emptyset$: one of their common vertices is in X.
(3) None of the above.

For $j \in \{1, 2, 3\}$, we denote $cost_j$ the sum of products of those overlapping districts of case (j). For case (1), using the given constraint that X is $(k, 5)$-orderly-scattered, we consider removing each set X_i one at a time from the graph in the increasing order of i. For each X_i, without loss of generality, we may swap the role of A and B such that for each summand we have $c_B \in X$. By applying Fact 11 (with $Y = X_i$), we know that for each district $A \in \mathcal{S}$, all districts B that overlap with A with $c_B \in X_i$ are actually centered at the same vertex. This implies that the sum of all such x_B values will be at most 1 by the primal constraint. Hence, the contribution of any district $A \in \mathcal{S}$ under case (1) for X_i in the graph $G - \cup_{j<i} X_j$ is at most

$$\sum_{B: A \cap B \neq \emptyset \text{ and } c_B \in X_i} x_A x_B \leq x_A.$$

By summing over all $A \in \mathcal{S}$ and over all the k sets $X_1, \ldots, X_k$, we have $cost_1 \leq k \cdot (\sum_S x_S)$. For case (2), the terms can be partitioned according to the common vertex c:

$$cost_2 \leq \sum_{j=1}^{k} \sum_{c \in X_j} \sum_{c \in A \cap B} x_A x_B \leq \sum_{j=1}^{k} \sum_{c \in X_j} \left(\sum_{c \in A} x_A \right)^2 \leq \sum_{j=1}^{k} \sum_{c \in X_j} \sum_{c \in A} x_A \leq k \cdot \left(\sum_{S} x_S \right).$$

Again here we use the property that for any fixed vertex c, the sum of the primal variables for star districts containing c sum up to be at most 1, i.e., $\sum_{c \in A} x_A \leq 1$. Further, fix an X_i, any star district includes at most one vertex from X_i.

For case (3) we can delegate the cost to the recursion. Notice that, all districts whose centers are in X will not participate in case (3). Hence, when considering each of the connected component in $G - X$, all the districts (after chopping off vertices in X) are still connected and are star-shaped.

Since X is a balanced separator, the divide and conquer analysis has at most $\log_{1/\delta} n$ layers. Thus, the sum over all products of overlapping districts is bounded by at most $2k \log_{1/\delta} n$ times the sum $\sum_S x_S$. ◀

5 FPTAS for General Districting on Complete Graphs and Trees

In this section, we present FPTAS for complete graphs and trees with weighted blocks. The algorithms here find c-balanced, connected districts that can be more than a star graph. Further, for complete graphs and trees, the LP-based algorithm in the previous section achieves $O(1)$-approximation ratio while the algorithms in this section achieves a ratio of $1 + \varepsilon$.

5.1 Complete Graph

Let G be a complete graph with functions of weights p_1 and p_2. Because we can merge two adjacent c-balanced districts on G into a single c-balanced district as shown in Fact 13, the c-balanced districting problem on complete graphs can be reduced to obtaining *one* c-balanced district, described as the following:

▶ **Fact 13** (Mergeable Property). *Assume T_1 and T_2 are disjoint districts and $G[T_1 \cup T_2]$ is connected. If T_1 and T_2 are both c-balanced, then $T_1 \cup T_2$ is also a c-balanced district.*

COMPLETE-GRAPH-c-BALANCED-DISTRICTING
Input: Let $G = (V, E)$ be a complete graph of n blocks and function of weights p_1 and p_2.
Goal: Obtaining a subset $S \subseteq V$ such that the total weight $w(S)$ is maximized subjected to the c-balanced condition:

$$(c-1)p_1(S) - p_2(S) \geq 0 \qquad \text{and} \qquad (c-1)p_2(S) - p_1(S) \geq 0. \tag{4}$$

The following theorem gives an FPTAS using dynamic programming (Algorithm 1).

Algorithm 1 FPTAS on complete graphs.

```
Input: ε > 0, c > 2, a complete graph (V, E), V = {v_1, ..., v_n}, functions of weights
       p = (p_1, p_2)
Function Trim(L, ℓ, ε):
    Sort L = {q_1, ..., q_m} so that ℓ(q_1) ≥ ℓ(q_2) ≥ ··· ≥ ℓ(q_m);
    Set L_out = ∅;
    for i = 1, ..., m do
        if q_i is not marked then
            L_out ← L_out ∪ {q_i};
            Mark all q_j ∈ L that ε-approximates q_i;
    return L_out;
Set L_1^0 = L_2^0 = {(0, 0)};
for i = 1, ..., n do
    L_1^i ← Trim(L_1^{i-1} ∪ (L_1^{i-1} + p(v_i)), ℓ_1, ε/n);
    L_2^i ← Trim(L_2^{i-1} ∪ (L_2^{i-1} + p(v_i)), ℓ_2, ε/n);
return the largest c-balanced districting in L_1^n ∪ L_2^n;
```

▶ **Theorem 14.** *There exists an FPTAS algorithm solving* COMPLETE-GRAPH-c-BALANCED-DISTRICTING *so that for all $c > 2$, $0 < \varepsilon < \frac{1}{2}\ln(c-1)$, and complete graph (V, E) of n nodes with functions of weights $\mathbf{p} = (p_1, p_2)$, the algorithm outputs an e^{ε}-approximation in $O\left(\varepsilon^{-4} n^6 (\ln w(V))^4\right)$ time where $w(V) = \sum_{v \in V} p_1(v) + p_2(v)$.*

We defer a detailed proof of Theorem 14 to Section C and here give a high level idea. One naive approach involves creating a complete list of potential subset sum values, denoted as $L(V)$ and outputting the largest c-balanced one. While this approach finds an optimal solution, it is not necessarily efficient, as $L(V)$ can be exponentially large. Similar to the knapsack problem or subset sum problem, one may use a bucketing idea to trim the list, keeping only one value when several are close to each other. However, the c-balanced constraint posts a challenge for the algorithm – for example, if an trimming algorithm keeps partial districts during the iterations, these partial districts may not always remain c-balanced resulting in a poor approximation ratio. To address this, we design a prioritized trimming process that prioritizes subsets satisfying the c-balanced condition in Equation (4) such that any c-balanced district in $L(V)$ would have an approximated district in our trimmed list.

Specifically, given $\varepsilon \geq 0$, we say $\mathbf{q}$ is an *ε-approximate* of $\mathbf{q}'$ if $q_1/q_1', q_2/q_2' \in [e^{-\varepsilon}, e^{\varepsilon}]$ where $0/0 := 1$. Let $\ell_1(\mathbf{q}) = (c-1)q_1 - q_2$ and $\ell_2(\mathbf{q}) = (c-1)q_2 - q_1$ be two linear functions on $\mathbf{q} = (q_1, q_2) \in \mathbb{R}^2$. We say $\mathbf{q}$ is *ℓ_j-dominated* by $\mathbf{q}'$ if $\ell_j(\mathbf{q}) \leq \ell_j(\mathbf{q}')$ for $j = 1, 2$, and $\mathbf{q}, \mathbf{q}' \in \mathbb{R}^2$, and L' is a *(ℓ_j, ε)-trimmed* of L if $L' \subseteq L$ and for each $\mathbf{q} \in L$ there exists $\mathbf{q}' \in L'$ which ε-approximates and ℓ_j-dominates $\mathbf{q}$. The key observation is that if $\mathbf{q}$ is c-balanced satisfying Equation (4) with $q_2 \geq q_1$ and $\mathbf{q}'$ ℓ_1-dominates $\mathbf{q}$, $\mathbf{q}'$ is also c-balanced. A similar argument holds for $q_1 \geq q_2$. This observation suggests that when trimming multiple nearby values, we keep the one that optimizes ℓ_1 (and ℓ_2) that ensures the existence of c-balanced approximated values. Therefore, we can find an e^{ε}-approximated solution if we can compute (ℓ_j, ε)-trimmed of all possible subset sum values $L(V)$. Moreover, because ℓ_1 and ℓ_2 are linear, we can use dynamic programming to sequentially and efficiently compute L_1^i and L_2^i that is $(\ell_1, \frac{\varepsilon i}{n})$-trimmed and $(\ell_2, \frac{\varepsilon i}{n})$-trimmed of all possible subset sum values on the first i blocks $L^i = L(\{v_1, \dots, v_i\})$ respectively. While Algorithm 1 only returns the size of our approximated solution $\mathbf{q} \in \mathbb{R}^2$, we can use an additional n factor to store the set S for each $\mathbf{q}$ in L_1^i, L_2^i to recover our approximated optimal districting.

Finally, we note that our prioritized trimming that ensures both inequalities in Equation (4): one through prioritized ℓ_j the other through exhausting cases of $q_2^* \geq q_1^*$ or $q_2^* \leq q_1^*$. However, we cannot extend this approach to non-binary color settings. Instead, if we allow relaxing c-balanced constraint to c'-balanced district with c' slightly larger than c, the standard bucketing algorithm mentioned above can directly work even for the non-binary color setting.

Adapting to the Star Districting Setting

We note that this dynamic programming approach also works for star districts on tree graph and yields an FPTAS. Similar to the arbitrary districts setting, we consider three cases for each v: the absent case where v is not included in any district; the consolidating where v is in a star district that is contained in its descendants; the incomplete case where v is the center of a star district that is incomplete.

References

1. Maxwell Allman, Itai Ashlagi, Irene Lo, Juliette Love, Katherine Mentzer, Lulabel Ruiz-Setz, and Henry O'Connell. Designing school choice for diversity in the San Francisco unified school district. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, EC '22, pages 290–291, New York, NY, USA, 2022. Association for Computing Machinery. doi:10.1145/3490486.3538271.
2. Micah Altman. Traditional districting principles: judicial myths vs. reality. *Social Science History*, 22(2):159–200, 1998.
3. Micah Altman and Michael P McDonald. Bard: Better automated redistricting. *Journal of Statistical Software*, 42:1–28, 2011.
4. Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta-algorithm and applications. *Theory of Computing*, 8(6):121–164, 2012. doi:10.4086/toc.2012.v008a006.

5 Per Austrin, Subhash Khot, and Muli Safra. Inapproximability of vertex cover and independent set in bounded degree graphs. *Theory Comput.*, 7(1):27–43, 2011. doi:10.4086/TOC.2011.V007A003.

6 Assaf Bar-Natan, Lorenzo Najt, and Zachary Schutzman. The gerrymandering jumble: map projections permute districts' compactness scores. *Cartography and Geographic Information Science*, 47(4):321–335, 2020.

7 Cristina Bazgan, Bruno Escoffier, and Vangelis Th. Paschos. Poly-APX- and PTAS-Completeness in standard and differential approximation. In Rudolf Fleischer and Gerhard Trippen, editors, *Algorithms and Computation*, pages 124–136, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

8 Amariah Becker and Justin Solomon. Redistricting algorithms. In Moon Duchin and Olivia Walch, editors, *Political Geometry: Rethinking Redistricting in the US with Math, Law, and Everything In Between*, pages 303–340. Springer International Publishing, Cham, 2022. doi:10.1007/978-3-319-69161-9_16.

9 Sayan Bhattacharya, Peter Kiss, and Thatchaphol Saranurak. Dynamic algorithms for packing-covering LPs via multiplicative weight updates. In Nikhil Bansal and Viswanath Nagarajan, editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 1–47. SIAM, 2023. doi:10.1137/1.9781611977554.CH1.

10 Sujoy Bhore, Sourav Chakraborty, Satyabrata Jana, Joseph S B Mitchell, Supantha Pandit, and Sasanka Roy. The balanced connected subgraph problem. *Discrete Appl. Math.*, 319:111–120, October 2022. doi:10.1016/J.DAM.2020.12.030.

11 Sujoy Bhore, Satyabrata Jana, Supantha Pandit, and Sasanka Roy. The balanced connected subgraph problem for geometric intersection graphs. *Theoretical Computer Science*, 929:69–80, 2022. doi:10.1016/J.TCS.2022.06.030.

12 Matteo Böhm, Adriano Fazzone, Stefano Leonardi, and Chris Schwiegelshohn. Fair clustering with multiple colors. arXiv 2002.07892, February 2020.

13 Victor Bucarey, Fernando Ordóñez, and Enrique Bassaletti. Shape and balance in police districting. In *Applications of location analysis*, pages 329–347. Springer, 2015.

14 Leonardo Bursztyn, Davide Cantoni, Patricia Funk, Felix Schönenberger, and Noam Yuchtman. Identifying the effect of election closeness on voter turnout: Evidence from Swiss referenda. *J. Eur. Econ. Assoc.*, 22(2):876–914, June 2023.

15 Timothy M. Chan and Sariel Har-Peled. Approximation algorithms for maximum independent set of pseudo-disks. *Discret. Comput. Geom.*, 48(2):373–392, 2012. doi:10.1007/S00454-012-9417-5.

16 Yuk Hei Chan and Lap Chi Lau. On linear and semidefinite programming relaxations for hypergraph matching. *Math. Program.*, 135(1-2):123–148, 2012. doi:10.1007/S10107-011-0451-5.

17 Yong Chen, Zhi-Zhong Chen, Guohui Lin, Yao Xu, and An Zhang. Approximation algorithms for maximally balanced connected graph partition. *Algorithmica*, 83(12):3715–3740, 2021. doi:10.1007/S00453-021-00870-3.

18 Anshuman Chhabra, Karina Masalkovaitė, and Prasant Mohapatra. An overview of fairness in clustering. *IEEE Access*, 9:130698–130720, 2021. doi:10.1109/ACCESS.2021.3114099.

19 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*, pages 5036–5044, 2017.

20 Miroslav Chlebík and Janka Chlebíková. Approximation hardness for small occurrence instances of NP-Hard problems. In *Algorithms and Complexity*, pages 152–164. Springer Berlin Heidelberg, 2003. doi:10.1007/3-540-44849-7_21.

21 Janka Chlebíková. Approximating the maximally balanced connected partition problem in graphs. *Information Processing Letters*, 60(5):225–230, 1996.

22 Gabriel Chuang, Oussama Hanguir, and Clifford Stein. Drawing Competitive Districts in Redistricting. In Guy N. Rothblum, editor, *5th Symposium on Foundations of Responsible*

Computing (FORC 2024), volume 295 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 7:1–7:22, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.FORC.2024.7.

23 Carmen Cirincione, Thomas A Darling, and Timothy G O'Rourke. Assessing South Carolina's 1990s congressional districting. *Political Geography*, 19(2):189–211, 2000.

24 S Clarke and J Surkis. An operations research approach to racial desegregation of school systems. *Socio-Economic Planning Sciences*, 1(3):259–272, 1968.

25 Jeanne Clelland, Haley Colgate, Daryl DeFord, Beth Malmskog, and Flavia Sancier-Barbosa. Colorado in context: Congressional redistricting and competing fairness criteria in Colorado. *Journal of Computational Social Science*, 5(1):189–226, 2022. doi:10.1007/S42001-021-00119-7.

26 Vincent Cohen-Addad, Philip N Klein, and Neal E Young. Balanced power diagrams for redistricting. *arXiv preprint arXiv:1710.03358*, 2017. arXiv:1710.03358.

27 Vincent Cohen-Addad, Philip N. Klein, and Neal E. Young. Balanced centroidal power diagrams for redistricting. In *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '18, pages 389–396, New York, NY, USA, 2018. Association for Computing Machinery. doi:10.1145/3274895.3274979.

28 Gerard Cornuéjols, David Hartvigsen, and William Pulleyblank. Packing subgraphs in a graph. *Operations Research Letters*, 1(4):139–143, 1982. doi:10.1016/0167-6377(82)90016-5.

29 Jessica De Silva, Brady Gales, Bryson Kagy, and David Offner. An analysis of a fair division protocol for drawing legislative districts. *arXiv preprint arXiv:1811.05705*, 2018.

30 Daryl DeFord, Moon Duchin, and Justin Solomon. A computational approach to measuring vote elasticity and competitiveness. *Statistics and Public Policy*, 7(1):69–86, 2020.

31 Daryl DeFord, Moon Duchin, and Justin Solomon. Recombination: A family of Markov chains for redistricting. *Harvard Data Science Review*, 3(1):3, 2021.

32 Guoli Ding and Wenan Zang. Packing cycles in graphs. *Journal of Combinatorial Theory, Series B*, 86(2):381–407, 2002. doi:10.1006/JCTB.2002.2134.

33 E. Falkenauer and A. Delchambre. A genetic algorithm for bin packing and line balancing. In *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 1186–1192 vol.2, 1992.

34 Allen D Franklin and Ernest Koenigsberg. Computed school assignments in a large district. *Operations Research*, 21(2):413–426, 1973. doi:10.1287/OPRE.21.2.413.

35 Roland G Fryer Jr and Richard Holden. Measuring the compactness of political districting plans. *The Journal of Law and Economics*, 54(3):493–535, 2011.

36 Zoltán Füredi, Jeff Kahn, and Paul D. Seymour. On the fractional matching polytope of a hypergraph. *Comb.*, 13(2):167–180, 1993. doi:10.1007/BF01303202.

37 Michael R Gary and David S Johnson. Computers and intractability: A guide to the theory of NP-completeness, 1979.

38 Nabeel Gillani, Doug Beeferman, Christine Vega-Pourheydarian, Cassandra Overney, Pascal Van Hentenryck, and Deb Roy. Redrawing attendance boundaries to promote racial and ethnic diversity in elementary schools. *Educational Researcher*, 52(6):348–364, 2023.

39 Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theor. Comput. Sci.*, 38:293–306, January 1985. doi:10.1016/0304-3975(85)90224-5.

40 Magnús Halldórsson and Jaikumar Radhakrishnan. Greed is good: approximating independent sets in sparse and bounded-degree graphs. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of Computing*, STOC '94, pages 439–448, New York, NY, USA, May 1994. Association for Computing Machinery. doi:10.1145/195058.195221.

41 J Gerald Hebert. *The Realist's Guide to Redistricting: Avoiding the Legal Pitfalls.* American Bar Association, 2010.

42 Sidney Wayne Hess, JB Weaver, HJ Siegfeldt, JN Whelan, and PA Zitlau. Nonpartisan political redistricting by computer. *Operations Research*, 13(6):998–1006, 1965.

43 Matt Jacobs and Olivia Walch. A partial differential equations approach to defeating partisan gerrymandering. *arXiv preprint arXiv:1806.07725*, 2018.
44 Xinrui Jia, Kshiteej Sheth, and Ola Svensson. Fair colorful k-center clustering. In *Integer Programming and Combinatorial Optimization*, pages 209–222. Springer International Publishing, 2020. doi:10.1007/978-3-030-45771-6_17.
45 Hai Jin. *Spatial optimization methods and system for redistricting problems*. PhD thesis, University of South Carolina, 2017.
46 Jörg Kalcsics, Stefan Nickel, and Michael Schröder. Towards a unified territorial design approach—applications, algorithms and GIS integration. *Top*, 13:1–56, 2005.
47 Peter Keevash and Benny Sudakov. Packing triangles in a graph and its complement. *Journal of Graph theory*, 47(3):203–216, 2004. doi:10.1002/JGT.20031.
48 Myung Jin Kim. *Optimization approaches to political redistricting problems*. PhD thesis, The Ohio State University, 2011.
49 Zeph Landau, Oneil Reid, and Ilona Yershov. A fair division solution to the problem of redistricting. *Social Choice and Welfare*, 32(3):479–492, 2009. doi:10.1007/S00355-008-0336-6.
50 Timothée Martinod, Valentin Pollet, Benoît Darties, Rodolphe Giroudeau, and J-C König. Complexity and inapproximability results for balanced connected subgraph problem. *Theoretical Computer Science*, 886:69–83, 2021. doi:10.1016/J.TCS.2021.07.010.
51 Elle Najt, Daryl DeFord, and Justin Solomon. Complexity and geometry of sampling connected graph partitions. *arXiv preprint arXiv:1908.08881*, 2019.
52 Gary Orfield and Danielle Jarvie. Black segregation matters: School resegregation and black educational opportunity. *Civil Rights Project-Proyecto Derechos Civiles*, 2020.
53 Wesley Pegden, Ariel D Procaccia, and Dingli Yu. A partisan districting protocol with provably nonpartisan outcomes. *arXiv preprint arXiv:1710.08781*, 2017. arXiv:1710.08781.
54 Serge A. Plotkin, David B. Shmoys, and Éva Tardos. Fast approximation algorithms for fractional packing and covering problems. In *32nd Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 1-4 October 1991*, pages 495–504. IEEE Computer Society, 1991. doi:10.1109/SFCS.1991.185411.
55 Alex Samorodnitsky and Luca Trevisan. A PCP characterization of NP with optimal amortized query complexity. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, STOC '00, pages 191–199, New York, NY, USA, 2000. Association for Computing Machinery. doi:10.1145/335305.335329.
56 Thomas C Schelling. Dynamic models of segregation. *Journal of mathematical sociology*, 1(2):143–186, 1971.
57 James B Weaver and Sidney W Hess. A procedure for nonpartisan districting: Development of computer techiques. *Yale Law Journal*, 73(2):288–308, 1963.
58 David Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. In Jon M. Kleinberg, editor, *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*, pages 681–690. ACM, 2006. doi:10.1145/1132516.1132612.

A Omitted Proofs from Section 3

Throughout, we refer type-1 vertices as vertices that have non-zero weight p_1 and zero weight of p_2 and type-2 vertices as vertices with non-zero weight p_2 and zero weight of p_1.

▶ **Theorem 4.** *The c-balanced districting problem does not have an $n^{1/2-\delta}$-approximation (for any constant $\delta > 0$) in a general graph unless $\mathsf{P} = \mathsf{NP}$. On a graph with maximum degree Δ, one cannot approximate the c-balanced districting problem within a factor of $\Delta/2^{O(\sqrt{\Delta})}$ assuming $\mathsf{P} \neq \mathsf{NP}$, and $O(\Delta/\log^2 \Delta)$ assuming the Unique Games Conjecture (UGC). Even if Δ is a constant, the problem is APX-hard. These statements hold when the districts must be stars and when the centers of the stars are limited to a subset of vertices.*

Proof. We take an instance of maximum independent set problem and turn it into an instance of c-balanced districting problem. Given a graph $G = (V, E)$, $n = |V|$ and $m = |E|$. Denote by Δ the maximum degree of G. For each vertex $v \in V$, we create a vertex v' in graph G' with type-1 weight of $(c-1)\Delta$ and type-2 weight of 0. There are $n = |V| = |V'|$ such "type-1 vertices". We also have a set of "type-2 vertices" V'' with type-2 weight of 1 and type-1 weight of 0. Each vertex $v' \in V'$ has exactly Δ type-2 neighbors. If u, v has an edge in G, in G', u' and v' share one type-2 vertex, which corresponds to the edge between u, v in G. See Figure 1. If the degree of u is less than Δ, the corresponding vertex in G' may have some dangling (degree-1) type-2 vertices. The total number of type-2 vertices is $n\Delta - m$. Thus the total number of vertices in G' is $n(\Delta + 1) - m$ and the number of edges in G' is $n\Delta$. In order for a type-1 vertex u' to be covered, all its type-2 neighbors must be used. Thus a maximum independent set S in G means we can cover all corresponding vertices of S in G' as well as all their type-2 neighbors, leading to a total coverage population of $|S|\Delta$. Similarly, if we can find a c-balanced districting problem in G', the type-1 vertices that are covered in c-balanced districts cannot share any common type-2 neighbors, and therefore the corresponding vertices in G must be independent. This reduction works when the district must be a star.

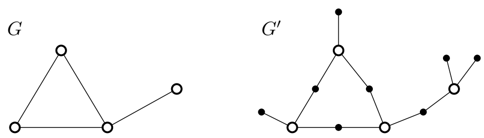

Figure 1 Graph G and G'. $\Delta = 3$ in this instance. The hallow vertices of G' have type-1 weight of $3(c-1)$ and the solid vertices of G' have type-2 weight of 1.

This reduction shows hardness of approximation, as an α-approximation for maximum independent set means an α-approximation for c-balanced districting problem, for any c. The maximum independent set cannot be approximated by a factor of $n^{1-\delta}$ for any constant $\delta > 0$ on general graph [7, 58]. If we have an approximation algorithm for the districting problem with approximation factor $O(N^{1/2-\delta})$ with N as the number of vertices in the districting graph G', by the reduction $N = O(n\Delta)$ and this gives an $O((n\Delta)^{1/2-\delta}) = O(n^{1-2\delta})$ for the maximum independent set problem on G, since $\Delta < n$, which is impossible unless $\mathsf{P} = \mathsf{NP}$.

As the maximum degree in both G and G' is Δ, approximating the balanced districting problem in G' with some factor depending on Δ gives the same approximation factor for the maximum independent set in G. For bounded degree graphs, the maximum independent set has a constant approximation [40], but is APX-complete [20] and cannot expect an approximation ratio better than $\Delta/2^{O(\sqrt{\Delta})}$ unless $\mathsf{P} = \mathsf{NP}$ [55]. Further, assuming the Unique Games Conjecture, one cannot approximate the maximum independent set problem within a factor of $O(\Delta/\log^2 \Delta)$ [5]. These (conditional) hardness results extend to balanced star districting problem on graphs with maximum degree Δ. This finishes the argument. ◀

B Separation Oracle and Randomized Rounding

B.1 Separation Oracle: Proof of Lemma 7

▶ **Lemma 7.** *Given an input instance $G = (V, E, (p_1, p_2))$, re-weighted values $w' : V \to \{0\} \cup [\frac{1}{w(G)}, w(G)]$, and dual variables $y'_v \in [n^{-(1+1/\varepsilon)}, 1+\varepsilon]$ for each vertex $v \in V$, there exists an algorithm that either reports that $\mathcal{S}_{\text{violate}} = \emptyset$, or returns a c-balanced district S such that $\sum_{v \in S} y'_v < (1-\varepsilon/2)w'(S)$ and $w'(S) \geq \frac{1}{2}w'(S_{\max})$, where $S_{\max} = \arg\max_{S \in \mathcal{S}_{\text{violate}}} w'(S)$ is a violating c-balanced district with the maximum value. The algorithm runs in $O(\varepsilon^{-6}n^6(\log n)(\log w(G))^4)$ time.*

Proof. We generalize Algorithm 1 to accommodate dual variables. In particular, we will maintain a candidate list L with the following property: For any c-balanced district S, there exists a district $T \in L$ such that (1) $(p_1(S), p_2(S))$ is an $(\varepsilon/10)$-approximate of $(p_1(T), p_2(T))$, and (2) $\sum_{v \in S} w'(v) / \sum_{v \in T} w'(v) \in [e^{-\varepsilon/10}, e^{\varepsilon/10}]$. Recall that from the proof of Theorem 14 we defined that a pair of numbers (q_1, q_2) is an ε-approximate to (q_1', q_2') if both $q_1/q_1', q_2/q_2' \in [e^{-\varepsilon}, e^{\varepsilon}]$. The above property suggests that we add a third dimension for y_v' to the list, and modify the trimming algorithm slightly – we will not trim the solution if their $\sum_{v \in S} y_v'$ values are too far from each other.

We now prove that the final list contains a c-balanced district that is a weakly violating constraint. Consider the population of commodities $(p_1(S_{\max}), p_2(S_{\max}))$ of $S_{\max}$. Without loss of generality, assume that $p_1(S_{\max}) \geq p_2(S_{\max})$. Then, by the property we stated above, there exists a district $S \in L$ that satisfies:

$$\begin{aligned}
(c-1)p_2(S) - p_1(S) &\geq (c-1)p_2(S_{\max}) - p_1(S_{\max}) && \text{(maintained by Algorithm 1)}\\
&\geq 0, \text{ and}\\
(c-1)p_1(S) - p_2(S) &\geq (1-\varepsilon/10)(c-1)p_1(S_{\max}) - (1+\varepsilon/10)p_2(S_{\max})\\
&\geq \left((1-\varepsilon/10)(c-2) - (\varepsilon/5)\right) p_2(S_{\max})\\
&\geq 0 && \text{(whenever } \varepsilon \leq \tfrac{c-2}{c}\text{)}
\end{aligned}$$

The above inequality shows that S is indeed c-balanced. Furthermore, we have $\sum_{v \in S} y_v' / \sum_{v \in S_{\max}} y_v' \in [e^{-\varepsilon/10}, e^{\varepsilon/10}]$. Thus, we are able to show that S is a weakly violating constraint:

$$\begin{aligned}
\sum_{v \in S} y_v' &\leq e^{\varepsilon/10} \cdot \sum_{v \in S_{\max}} y_v'\\
&\leq e^{\varepsilon/10} \cdot (1-\varepsilon) \cdot w'(S_{\max}) && (S_{\max} \text{ is strongly violating})\\
&\leq e^{\varepsilon/10} \cdot (1-\varepsilon) \cdot e^{\varepsilon/10} \cdot w'(S)\\
&\leq e^{\varepsilon/5} \cdot (1-\varepsilon) \cdot w'(S) \leq (1-\varepsilon/2) \cdot w'(S). && (\varepsilon > 0)
\end{aligned}$$

On the other hand, we have

$$w'(S) \geq e^{-\varepsilon/10} \cdot w'(S_{\max}) \geq \frac{1}{2} w'(S_{\max}),$$

certifying that the output S satisfies all the constraints from the lemma statement.

Let us now analyze the runtime of the algorithm. It suffices to analyze the number of scales at the new dimension. Since each value y_v' is at least $n^{-(1+1/\varepsilon)}$ and is at most $1+\varepsilon$, the number of scales in the third dimension can be bounded by

$$\log_{e^{\varepsilon/10}} \frac{1+\varepsilon}{n^{-(1+1/\varepsilon)}} = \frac{\ln(1+\varepsilon) + (1+1/\varepsilon)\ln n}{\varepsilon/10} = O(\varepsilon^{-2} \ln n).$$

Together with the analysis in Theorem 14, the runtime of this generalized algorithm for the complete graph, including maintaining a solution, is $O(\varepsilon^{-6} n^6 (\log n)(\log w(G))^4)$. ◀

B.2 Proof of Randomized Rounding

▶ **Lemma 15.** *The randomized rounding algorithm from the fractional LP solution produces an expected weight for output districting I as*

$$\mathbf{E}[w(I)] \geq \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) \frac{x_S}{\tau} - \sum_{A,B \in \mathcal{S}_{\mathrm{LP}}:\ A \cap B \neq \emptyset} \min(w(A), w(B)) \frac{x_A x_B}{\tau^2}\ .$$

Proof. Consider a district S with non-zero value x_S, it is selected into I only if two events happen: (1) the coin flip with probability x_S/τ turns out to be true; and (2) all the districts with value at least x_S are not included in I – their coin flips are false. The probability of both events happening is

$$\frac{x_S}{\tau} \cdot \prod_{A \in \mathcal{S}_{\mathrm{LP}}: A \cap S \neq \emptyset, w(A) \geq w(S)} \left(1 - \frac{x_A}{\tau}\right) \geq \frac{x_S}{\tau} \cdot \left(1 - \sum_{A \in \mathcal{S}_{\mathrm{LP}}: A \cap S \neq \emptyset, w(A) \geq w(S)} \frac{x_A}{\tau}\right)$$

Now, by linearity of expectation, we have

$$\begin{aligned}
\mathbf{E}[w(I)] &\geq \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) \frac{x_S}{\tau} \cdot \left(1 - \sum_{A \in \mathcal{S}_{\mathrm{LP}}: A \cap S \neq \emptyset, w(A) \geq w(S)} \frac{x_A}{\tau}\right) \\
&= \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) \cdot \frac{x_S}{\tau} - \sum_{S, A \in \mathcal{S}_{\mathrm{LP}}: A \cap S \neq \emptyset, w(A) \geq w(S)} w(S) \cdot \frac{x_S x_A}{\tau^2} \\
&= \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) \cdot \frac{x_S}{\tau} - \sum_{A, B \in \mathcal{S}_{\mathrm{LP}}:\ A \cap B \neq \emptyset} \min(w(A), w(B)) \cdot \frac{x_A x_B}{\tau^2}\ .
\end{aligned}$$ ◀

For any $\delta \geq 0$, let $\mathcal{S}_{\geq \delta}$ be the set of all districts $S \subseteq \mathcal{S}_{\mathrm{LP}}$ whose weight is at least δ. The following lemma connects the unweighted correlation between overlapped districts and the expected approximation ratio to the randomized rounding algorithm.

▶ **Lemma 16.** *Let $\tau \in \mathbb{R}_{>0}$ be a fixed value. Suppose that for all $\delta > 0$,*

$$\sum_{A, B \in \mathcal{S}_{\geq \delta}: A \cap B \neq \emptyset} x_A x_B \leq (\tau/2) \cdot \sum_{S \in \mathcal{S}_{\geq \delta}} x_S, \tag{5}$$

then $\mathbf{E}[w(I)] \geq \frac{1}{2\tau} \sum_S w(S) \cdot x_S$.

Proof. We first sort all districts in $\mathcal{S}_{\mathrm{LP}}$ in the non-increasing order of weights. Let $S_1, S_2, \ldots, S_t$ be such a list. For each district S_i, its weight $w(S_i)$ can be written as

$$w(S_i) = \sum_{j=i}^{t} (w(S_j) - w(S_{j+1})).$$

Here for convenience we define $w(S_{t+1}) = 0$. Using the above expression, we are able to establish that

$$\begin{aligned}
&\sum_{A, B \in \mathcal{S}_{\mathrm{LP}}:\ A \cap B \neq \emptyset} \min(w(A), w(B)) \cdot \frac{x_A x_B}{\tau^2} \\
&= \sum_{i=1}^{t} \sum_{\ell=1}^{i-1} \mathbb{I}[S_\ell \cap S_i \neq \emptyset] \cdot w(S_i) \cdot \frac{x_{S_i} x_{S_\ell}}{\tau^2} \\
&= \sum_{i=1}^{t} \sum_{\ell=1}^{i-1} \mathbb{I}[S_\ell \cap S_i \neq \emptyset] \cdot \left(\sum_{j=i}^{t} (w(S_j) - w(S_{j+1}))\right) \cdot \frac{x_{S_i} x_{S_\ell}}{\tau^2} \\
&= \sum_{j=1}^{t} \Big(w(S_j) - w(S_{j+1})\Big) \cdot \left(\sum_{i=1}^{j} \sum_{\ell=1}^{i-1} \mathbb{I}[S_\ell \cap S_i \neq \emptyset] \cdot \frac{x_{S_i} x_{S_\ell}}{\tau^2}\right) \\
&= \sum_{j=1}^{t} \Big(w(S_j) - w(S_{j+1})\Big) \cdot \left(\frac{1}{\tau^2} \cdot \sum_{A, B \in \mathcal{S}_{\geq w(S_j)}:\ A \cap B \neq \emptyset} x_A x_B\right)
\end{aligned}$$

$$\le \sum_{j=1}^{t} \Big(w(S_j) - w(S_{j+1})\Big) \cdot \left(\frac{1}{\tau^2} \cdot \frac{\tau}{2} \cdot \sum_{S \in \mathcal{S}_{\ge w(S_j)}} x_S\right) \qquad \text{(by (5))}$$

$$= \frac{1}{2\tau} \sum_{j=1}^{t} \sum_{i=1}^{j} \Big(w(S_j) - w(S_{j+1})\Big) \cdot x_{S_i}$$

$$= \frac{1}{2\tau} \sum_{i=1}^{t} x_{S_i} \cdot \sum_{j=i}^{t} \Big(w(S_j) - w(S_{j+1})\Big)$$

$$= \frac{1}{2\tau} \sum_{i=1}^{t} w(S_i) \cdot x_{S_i}$$

Finally, we have

$$\begin{aligned} \mathbf{E}[w(I)] &\ge \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) \frac{x_S}{\tau} - \sum_{A,B \in \mathcal{S}_{\mathrm{LP}}:\ A \cap B \ne \emptyset} \min(w(A), w(B)) \frac{x_A x_B}{\tau^2} \qquad \text{(by Lemma 15)} \\ &\ge \frac{1}{\tau} \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) x_S - \frac{1}{2\tau} \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) x_S \\ &= \frac{1}{2\tau} \sum_{S \in \mathcal{S}_{\mathrm{LP}}} w(S) x_S \end{aligned}$$

as desired. ◀

C Omitted Proofs from Section 5

▶ **Theorem 14.** *There exists an FPTAS algorithm solving* Complete-Graph-c-Balanced-Districting *so that for all $c > 2$, $0 < \varepsilon < \frac{1}{2}\ln(c-1)$, and complete graph (V, E) of n nodes with functions of weights $\mathbf{p} = (p_1, p_2)$, the algorithm outputs an e^{ε}-approximation in $O\left(\varepsilon^{-4} n^6 (\ln w(V))^4\right)$ time where $w(V) = \sum_{v \in V} p_1(v) + p_2(v)$.*

Proof of Theorem 14. Given an arbitrary ordering on blocks, let L^i be the set of all values that can be obtained by selecting some subset of the first i blocks $\{v_1, \dots, v_i\}$,

$$L^i = \left\{ \exists S \subseteq [i], \sum_{j \in S} \mathbf{p}(v_j) \right\} \subset \mathbb{Z}^2_{\ge 0}.$$

We use induction to show that L_1^i in Algorithm 1 is an $(\ell_1, \frac{\varepsilon i}{n})$-trimmed of L^i for all i. The base case $i = 0$ is trivially holds as $L^0 = \emptyset$. Suppose L_1^{i-1} is an $\left(\ell_1, \frac{\varepsilon(i-1)}{n}\right)$-trimmed of L^{i-1}. For any $\mathbf{q} + \mathbf{p}(v_i) \in L^i \setminus L^{i-1}$ with $\mathbf{q} \in L^{i-1}$, by the induction hypothesis, there exists $\mathbf{q}' \in L_1^{i-1}$ which $\frac{\varepsilon(i-1)}{n}$-approximates and ℓ_1-dominates $\mathbf{q}$. Because $p_1(v_i), p_2(v_i) \ge 0$,

$$\frac{q_1' + p_1(v_i)}{q_1 + p_1(v_i)}, \frac{q_2' + p_2(v_i)}{q_2 + p_2(v_i)} \in [e^{\frac{-\varepsilon(i-1)}{n}}, e^{\frac{\varepsilon(i-1)}{n}}] \text{ and } \ell_1(\mathbf{q}' + \mathbf{p}(v_i)) \ge \ell_1(\mathbf{q} + \mathbf{p}(v_i)).$$

On the other hand, by the definition of L_1^i, for any $\mathbf{q}' + \mathbf{p}(v_i) \in L_1^{i-1} + \mathbf{p}(v_i)$ there exists $\mathbf{q}'' \in L_1^i$ so that

$$\frac{q_1''}{q_1' + p_1(v_i)}, \frac{q_2''}{q_2' + p_2(v_i)} \in [e^{\frac{-\varepsilon}{n}}, e^{\frac{\varepsilon}{n}}] \text{ and } \ell_1(\mathbf{q}'') \ge \ell_1(\mathbf{q}' + \mathbf{p}(v_i))$$

Combining these two proves that $\mathbf{q}''$ $\frac{\varepsilon i}{n}$-approximates and ℓ_1-dominates $\mathbf{q} + \mathbf{p}(v_i)$. The identical argument holds for all $\mathbf{q} \in L^{i-1} \subseteq L^i$. Thus, we show L_1^i is an $(\ell_1, \frac{\varepsilon i}{n})$-trimmed of L^i for all i. Similar argument applies to L_2^i.

Let $\mathbf{q}^*$ be the optimal c-balanced value in L^n. Suppose $q_2^* \geq q_1^*$. Since L_1^n is (ℓ_1, ε)-trimmed to L^n, there exists $\mathbf{q}' \in L_1^n$ that ε-approximates and ℓ_1-dominates $\mathbf{q}^*$. Because $\mathbf{q}'$ ε-approximates $\mathbf{q}^*$, the approximation guarantee holds, $q_1' + q_2' \geq e^{-\varepsilon}(q_1^* + q_2^*)$. Now we show $\mathbf{q}'$ is also c-balanced. Because $\mathbf{q}^*$ is c-balanced and $\mathbf{q}'$ ℓ_1-dominates $\mathbf{q}^*$,

$$0 \leq (c-1)q_1^* - q_2^* = \ell_1(\mathbf{q}^*) \leq \ell_1(\mathbf{q}').$$

Moreover, because $q_2^* \geq q_1^*$ and $\mathbf{q}'$ ε-approximates $\mathbf{q}^*$, we have

$$(c-1)q_2' \geq (c-1)e^{-\varepsilon}q_2^* \geq (c-1)e^{-\varepsilon}q_1^* \geq (c-1)e^{-2\varepsilon}q_1' \geq q_1'$$

where the last inequality holds because $\frac{1}{2}\ln(c-1) \geq \varepsilon$. Combining these two, we have $\ell_1(\mathbf{q}')$ and $\ell_2(\mathbf{q}') \geq 0$ completing the proof. Similarly, if $q_2^* \geq q_1^*$, there exists an ε-approximation and c-balanced solution in L_2^n.

The running time of i-th iteration is $O(|L_1^i|^2 + |L_2^i|^2)$ which can be bounded as the following. Consider a geometric grid with vertices in $\{(e^{\frac{j}{n}\varepsilon}, e^{\frac{k}{n}\varepsilon}) : j, k = 0, \ldots, \lceil \frac{n}{\varepsilon} \ln w(V) \rceil\}$. Because $L_1^i \subseteq [w(V)]^2$ and no two points in can be in a same rectangle after trimming, the size of L_1^i is bounded by the size of grid $O(\frac{n^2}{\varepsilon^2}(\ln w(V))^2)$. Therefore, the running time of Algorithm 1 is $O(\frac{n^5}{\varepsilon^4}(\ln w(V))^4)$. The additional n in the theorem statement is to reconstruct the set. ◀