Gerrymandering Planar Graphs

Jack Dippel
McGill University
jack.dippel@mail.mcgill.ca

Max Dupré la Tour
McGill University
max.duprelatour@mail.mcgill.ca

April Niu
McGill University
yuexing.niu@mail.mcgill.ca

Sanjukta Roy*
University of Leeds
s.roy@leeds.ac.uk

Adrian Vetta
McGill University
adrian.vetta@mcgill.ca

ABSTRACT

We study the computational complexity of the map redistricting problem (gerrymandering). Mathematically, the electoral district designer (gerrymanderer) attempts to partition a weighted graph into k connected components (districts) such that its candidate (party) wins as many districts as possible. Prior work has principally concerned the special cases where the graph is a path or a tree. Our focus concerns the realistic case where the graph is planar. We prove that the gerrymandering problem is solvable in polynomial time in λ-outerplanar graphs, when the number of candidates and λ are constants and the vertex weights (voting weights) are polynomially bounded. In contrast, the problem is NP-complete in general planar graphs even with just two candidates. This motivates the study of approximation algorithms for gerrymandering planar graphs. However, when the number of candidates is large, we prove it is hard to distinguish between instances where the gerrymanderer cannot win a single district and instances where the gerrymanderer can win at least one district. This immediately implies that the redistricting problem is inapproximable in polynomial time in planar graphs, unless P=NP. This conclusion appears terminal for the design of good approximation algorithms – but it is not. The inapproximability bound can be circumvented as it only applies when the maximum number of districts the gerrymanderer can win is extremely small, say one. Indeed, for a fixed number of candidates, our main result is that there is a constant factor approximation algorithm for redistricting unweighted planar graphs, provided the optimal value is a large enough constant.

KEYWORDS

Social Choice Theory; Redistricting; Gerrymandering; Approximation Algorithm; Planar Graph

ACM Reference Format:
Jack Dippel, Max Dupré la Tour, April Niu, Sanjukta Roy, and Adrian Vetta. 2024. Gerrymandering Planar Graphs. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 9 pages.

*The author was affiliated to Pennsylvania State University when the work was done.

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand.

1 INTRODUCTION

Partisan gerrymandering refers to the manipulation of district lines in order to give an advantage to one political party over others. This is facilitated by the hierarchical structure of political elections which allows for manipulation at the district or division level to ultimately secure more seats. An effective strategy that divides the voting population can lead to a party or candidate winning a higher number of seats than they would otherwise have won. This process is commonly referred to as *gerrymandering*. The US House of Representatives elections provide a clear example of this practice, where candidates from different parties compete at the district level to represent their constituents in Congress. The problem of gerrymandering has recently gained considerable attention in the computer science community. Algorithms are used in practice to create districts (for example, many states within Mexico [25] use computer algorithms to district) and also to test for gerrymandering. The discussion of algorithmically finding optimal districts started in the '60s [1, 18]. This induces the algorithmic question of whether it is possible to redistrict in such a way as to secure a greater number of seats for the favored political party (the *gerrymanderer*).

A ubiquitous constraint in gerrymandering is that the counties that form a district be contiguous. Thus gerrymandering is simply a graphical problem, where a vertex represents a county and an edge denotes the existence of a shared border between the two corresponding counties. The task then is to partition the graph into connected subgraphs, namely districts, such that one specific candidate (or party) will win as many districts as possible. This problem, called *gerrymandering over graphs* was introduced by Cohen-Zemach, Lewenberg, and Rosenschein [9]. The computational complexity of the gerrymandering problem has since received considerable attention in the literature. Clearly, planar graphs capture the setting of interest in practice, namely redistricting a geographical area. However, prior work on the gerrymandering problem has predominantly focused upon the special case of networks that are either paths or trees [4, 6, 9, 16, 20, 24]. But what can we say about real instances, specifically, what happens in maps? Answering this question motivates our study of gerrymandering on *planar* graphs.

1.1 The Gerrymandering Model

An election with a set C of c candidates is modeled by a graph $G = (V, E)$ where vertices represent counties and edges indicate geographic adjacency. The *gerrymandering* problem of Cohen-Zemach et al. [9] is then as follows. There is a vertex weight function $wt\colon V(G) \to \mathbb{N}$ and an approval function $a : V(G) \to C$, where

$a(v)$ represents the candidate approved of by v. Given a distinguished candidate $i \in C$ and a connected subgraph $T \subseteq V(G)$, we say that candidate i *wins district* T if $\sum_{v \in T,\, a(v)=i} wt(v) > \sum_{v \in T,\, a(v)=j} wt(v)$, for any other candidate j.

> The Gerrymandering Problem: Given two positive integers k and w, and a candidate b, is there a partition of $V(G)$ into k non-empty, connected districts $T_1 \uplus \ldots \uplus T_k$ such that candidate b wins at least w districts?

We will also consider a *bounded district version* of the problem where we incorporate lower and upper bounds on the number of counties in any district (or lower and upper bounds on the total weight of the vertices of any district). That is, $\ell \leq |T_i| \leq u$ for some $1 \leq \ell \leq u \leq |V(G)|$ and each district T_i.

A feasible solution is called a (k, w)*-partition*. The decision problem is to determine whether a (k, w)-partition exists. The corresponding optimization problem is to find the largest possible w_{OPT} such that a (k, w_{OPT})-partition exists.

Alternately, we can represent each candidate by a color, with the gerrymanderer b being denoted by "blue". Each vertex is given the color of its preferred candidate. Given a partition of the graph into k connected components (districts), blue wins a district if it has the strictly highest total weight of any color in the district. The gerrymanderer desires a partition that maximizes the number of districts won by blue. We say a district is a *winning district* if blue wins it.

1.2 Our Contribution

Our focus is upon the natural restriction that the graph G is planar. However, Ito et al. [20] showed the gerrymandering problem is NP-complete, even if the number of districts is $k = 2$, the number of candidates is $c = 2$, and G is the bipartite graph $K_{2,n}$. Since $K_{2,n}$ is planar, the problem is hard for planar graphs. In this paper we present much stronger hardness results and initiate the study of approximation algorithms for the gerrymandering problem in planar graphs. Moreover, we provide exact algorithms for a slight restriction of planar graphs.

Exact solutions. Redistricting graphs is shown to be NP-hard even in restricted graph classes [4, 14, 16] and there have been a few algorithmic results, mainly for simple structures such as paths [16] and trees [4, 14]. We push the boundaries of known complexity results. In Section 2, we study λ-outerplanar graphs. A graph G is *λ-outerplanar* if G has a planar embedding such that the vertices belong to λ layers. Define L_1 to be the vertices incident to the outer-face, and define L_i for $i > 1$ recursively to be the vertices on the outer-face of the planar drawing obtained by removing the vertices in $L_1, L_2, \ldots, L_{i-1}$. Each L_i is called a layer.[1] We remark that λ-outerplanar graphs with a constant index λ are important as the graphs that arise in practical maps have this property. We prove that, for constant λ and a constant number of candidates, there is a polynomial time algorithm to solve the gerrymandering problem for λ-outerplanar graphs. This positive result (Theorem 2.2) applies to the districting model in its full generality, including with lower and upper bounds on the district sizes, provided each vertex weight is polynomially bounded. Note that this result is tight; that is, unless P=NP, we cannot remove the dependency on the weight in the running time since it is NP-hard for paths with general weights [4]. To show Theorem 2.2 we, in fact, prove a more general result (Theorem 2.1): the gerrymandering problem is solvable in polynomial time in graphs of bounded treewidth, given a constant number of candidates and polynomial vertex weights. In particular, our result extends the approach of [20] on trees to graphs of bounded treewidth.

Hardness of approximation. In Section 3 we explore the hardness of the gerrymandering problem in more detail. First, we consider the case of two candidates and general graphs. We show, via a reduction from the independent set problem, that the gerrymandering problem is inapproximable within an $O(n^{\frac{1}{3}-\epsilon})$ factor, for any constant $\epsilon > 0$ (Theorem 3.2). This result holds even for the *unweighted* case where each vertex has weight one. We remark this result only implies NP-hardness for planar graphs and not inapproximability because there exists a polynomial-time approximation scheme (PTAS) for independent set in planar graphs [3]. Much stronger results arise with many candidates. Specifically, for planar graphs it is hard to distinguish between instances where the gerrymanderer cannot win a single district and instances where the gerrymanderer can win at least one district (Theorem 3.3). This implies that the gerrymandering problem is inapproximable in polynomial time in planar graphs, even in the unweighted case!

Approximate solutions. This hardness result suggests no approximation algorithm is possible for planar graphs. But the situation is more subtle. It actually implies no approximation is possible if the optimal solution w_{OPT} is small, for example if $w_{OPT} = 1$. Remarkably, in Section 4, we show that for a constant number of candidates there is a constant factor approximation algorithm in unweighted planar graphs when w_{OPT} is a large enough constant. Specifically, we present a quasi-linear time $O(c)$-approximation algorithm provided the optimal number of winning districts is a sufficient multiple of the number of candidates c (Theorem 4.1).

In Section 5, we study a problem called the *singleton winning district gerrymandering problem* where the gerrymanderer aims to win only singleton districts. This combinatorially interesting but seemingly contrived problem is significant because its approximability relates closely to the approximability of the generic gerrymandering problem. Combining our algorithm for λ-outerplanar graphs with Baker's method [3] for planar graphs, we prove that there is a polynomial time approximation scheme (PTAS) for the unweighted singleton winning district gerrymandering problem in planar graphs (Theorem 5.1). This raises an intriguing structural question that we leave open: does a PTAS also exist for the (non-singleton winning) gerrymandering problem, again assuming w_{OPT} is large enough?

1.3 Background and Related Work

For many decades, the existence of gerrymandering and its consequences have been widely acknowledged and discussed in the realm of political science, as documented by Erikson [12], Lubin [22], and Issacharoff [19]. However, the practical feasibility and broad implications of gerrymandering have only recently become the subject of intense public, policy, and legal debate [26], largely due to the widespread use of computer modeling in the election process.

[1] Equivalently, iteratively deleting the outer layer (unbounded face) of the graph will produce an empty graph in λ iterations.

Computational complexity studies of gerrymandering have taken several forms. Puppe and Tasnádi [23] explored gerrymandering under certain constraints where specific groups of voters cannot be included in the partition and showed it is NP-complete. Fleiner et al. [13], Lewenberg et al. [21] and Eiben et al. [11] investigated gerrymandering in presence of geographical constraints.

Most closely related to our work, there has been extensive recent study on gerrymandering graphs. The model was originally introduced by Cohen-Zemach et al. [9]. They showed it is NP-complete to decide if there is a (k, w)-partition of a weighted graph, in the case where districts of size 1 are not allowed. They also designed a greedy algorithm for gerrymandering over graphs and analyzed its performance empirically through simulations on random graphs. Ito et al. [20] extended those results by considering a slightly different model, where the goal is to win more districts than any other candidate in a global election. We refer to this model as the *global election model.* They proved that the problem is NP-complete for several graph classes: complete graphs, $K_{2,n}$, unweighted graphs with $c = 4$, and trees of diameter four. They also provided polynomial-time algorithms for stars and pseudo-polynomial time algorithms for paths and trees when c is constant. Gupta et al. [16] extended the model of gerrymandering to include *vector weighted vertices*, where a vertex can have different weights for different candidates. They proved, in the global election model, that the problem of gerrymandering is NP-complete on paths. They also presented algorithms with running times of $2^k(n + c)^{O(1)}$ and $2^n(n + c)^{O(1)}$ for paths and general graphs, respectively. Bentert et al. [4] considered the global election model with trees only and proved NP-hardness for the problem on paths, even in the unweighted case. They also provided a polynomial-time algorithm for trees when $c = 2$ and proved weakly NP-hardness for trees when $c \geq 3$. They further presented a polynomial-time algorithm for trees of diameter exactly three. Furthermore, Fraser et al. [14] considered the global election model on trees. They proved that the problem is W[2]-hard when parameterized by the number of districts k, for trees of depth two. They also provided an algorithm with running time $O(n^{3l} \cdot 2.619^k(n+m)^{O(1)})$ for the vector weighted case on trees where l is the number of leaves. Graph theoretic formulation of districting problem has been used in [2, 7] to study structural properties of the problem on grids.

Gerrymandering over graphs has also been studied from the complementary viewpoint of fairness as opposed to gerrymandering. Stoica et al. [24] studied gerrymandering graphs with an objective of creating fair connected districts: where the maximum margin of victory of any candidate is minimized, and showed NP-hardness for $k = 2$ and $c = 2$. Boehmer et al. [6] showed W[1]-hardness and an XP algorithm parameterized by treewidth, k and c. We emphasize our algorithm for gerrymandering in graphs of bounded treewidth is independent of k.

2 A POLYNOMIAL TIME ALGORITHM FOR GRAPHS WITH BOUNDED TREEWIDTH

In this section, we introduce a dynamic program approach that efficiently computes an exact solution in polynomial time, provided that the treewidth of the graph and the number of candidates are constant. It is motivated by algorithm for trees of Ito et al. [20]. Here, the graph is weighted, that is, each vertex represents multiple votes for a candidate. Additionally, it allows for other constraints such as incorporating lower and upper bounds on district sizes.

Theorem 2.1. *There exists an algorithm computing the maximum number of winning district in time* $O(n^{2tw+7} \cdot (\sum_{v \in V} wt(v))^{2c \cdot tw})$, *where* tw *is the treewidth of the graph.*

We defer the proof of Theorem 2.1 and technical definitions required for the proof to the full paper [10]. We remark that Theorem 2.1 is tight in the sense that we cannot eliminate the dependency on c. This is because the problem is NP-complete for paths (whose treewidth is one), a result due to Bentert et al. [4].[2] Moreover, it is impossible to omit the vertex-weight term and obtain an algorithm with running time $n^{O(c \cdot tw)}$, unless P=NP, since the problem is weakly NP-hard even if $tw = 1$ (namely, trees) and $c = 3$ [4].

We further remark that if the graph G is λ-outerplanar, that is, G has a planar embedding such that the vertices belong to λ layers, then G has treewidth at most $3\lambda - 1$; see Bodlaender [5]. Thus λ-outerplanar graphs have bounded treewidth if λ is constant. So our dynamic program can be applied to the practical setting of λ-outerplanar graphs. Specifically, we obtain the following result as a corollary of Theorem 2.1.

Theorem 2.2. *In a* λ*-outerplanar graph* G*, there is an algorithm to compute the maximum number of winning districts in time* $n^{O(c\lambda)}$ *if the vertex weights are polynomially bounded.*

3 ON THE HARDNESS OF THE GERRYMANDERING PROBLEM

It is known that gerrymandering is NP-hard for unweighted paths when there are many candidates [4]. We now present much stronger inapproximability results for unweighted (planar) graphs.

3.1 Lower Bounds for Elections with Two Candidates

To begin, we consider the natural setting where there are two candidates (or parties) in the election. We show that the gerrymandering problem is at least as hard as the *maximum independent set problem.* Proofs for the next two results can be found in full paper.

Lemma 3.1. *Given a graph* G *on* n *vertices and an integer* w*, there exists a graph* H *of size polynomial in* n *and a coloration of* H *using two colors such that there exists a* w*-sized independent set in* G if and only if *there exists a* (n, w)*-partition in* H.

Using this link to the independent set problem [17, 27] we can obtain a very strong inapproximability bound for the gerrymandering problem with two candidates in general graphs.

Theorem 3.2. *In gerrymandering with two candidates, for any* $\epsilon > 0$*, there is no polynomial-time algorithm with approximation guarantee less than* $n^{1/3-\epsilon}$*, unless P = NP.*

3.2 Lower Bounds for Multi-Candidate Elections in Planar Graphs

We remark that, for a planar graph G, the reduction in Lemma 3.1 produces a planar auxiliary graph H. Thus, Lemma 3.1 implies that,

[2] The result is for the global election model but it applies for the model [9] we study.

for two candidates the gerrymandering problem is NP-complete in planar graphs. But, as we shall now see, when there is a large number of candidates the situation is far worse: the gerrymandering problem is inapproximable, even in planar graphs! This is a consequence of the next result.

Theorem 3.3. *For planar graphs, finding a partition with one winning district is NP-hard.*

Proof. We construct a reduction from the *minimum connected vertex cover problem*. Here, given a connected graph G, we seek a vertex cover $S \subseteq V(G)$ of minimum cardinality such that the subgraph induced by S is connected. This problem is NP-hard even in planar graphs of maximum degree four [15]. Now, given an integer l, we build an auxiliary graph H from G such that there is a size l connected vertex cover in G *if and only if* there is an $(n, 1)$-partition of H. The construction of H, shown in Figure 1, follows. For every vertex $v \in V(G)$:

- There is a white vertex $w^v \in V(H)$.
- There are $n|E| \cdot (|E| + 1)$ red vertices $r_i^v \in V(H)$, for $i = 1, \ldots, n|E| \cdot (|E| + 1)$. Each such red vertex has an edge in $E(H)$ to w^v.

For every edge $e = (u, v) \in E(G)$:

- There is a blue vertex $b^e \in V(H)$ with edges to both w^u and w^v.
- There are $nl \cdot (|E| + 1)$ blue vertices $b_i'^e \in V(H)$, for $i = 1, \ldots, nl \cdot (|E| + 1)$. Each such blue vertex has an edge in $E(H)$ to b^e.
- For all $b_i'^e$, $i \in \{1, \ldots, nl(|E| + 1)\}$, there are $|E|$ vertices $c_{i,j}^e$, for $j = 1, \ldots |E|$, each with an edge to $b_i'^e$. All vertices of the form $c_{i,j}^e$ are all colored with the same new color C^e.

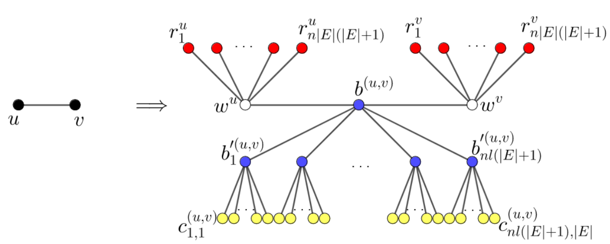

Figure 1: Construction of H in the proof of Theorem 3.3

Observe that the transformation from G to H preserves planarity. Consider the gerrymandering problem on H with n districts. First, we show that if there is a connected vertex cover S in G of size at most l then there is a decomposition of H with one winning district W. For all $v \notin S$, we create a district D_v with the vertex w^v and all the vertices of the form r_i^v for $i \in \{1, \ldots, n|E| \cdot (|E| + 1)\}$. All the remaining vertices are placed in the district W. Observe that W is connected because S is a connected vertex cover. In W there are:

- $|E| \cdot nl \cdot (|E| + 1) + |E|$ blue vertices;
- $|S|$ white vertices;
- $|S| \cdot n|E| \cdot (|E| + 1)$ red vertices;
- $|E| \cdot nl \cdot (|E| + 1)$ vertices of color C^e, for each $e \in E$.

So W is winning because it contains more blue than red vertices, as $|S| \leq l$.

For the other direction, assume that W is a winning district in H. Define E^W to be the set of edges $e \in E(G)$ such that $b^e \in W$ and define V^W to be the set of vertices $v \in V$ such that $w^v \in W$. We claim V^W is a connected vertex cover of cardinality at most l in G. To show this, observe that V^W is connected in G because W is connected in H. Next, to show that V^W is a vertex cover, it suffices to prove that $E^W = E$. Suppose, for the sake of contradiction, that $|E^W| < |E|$. Recall, there are only n districts in total and each district must be connected. So if $b^{(u,v)} \in W$ for some $(u, v) \in E(G)$, then at most n vertices of the form $b_i'^{(u,v)}$ are not in W for $i \in \{1, \ldots, nl(|E| + 1)\}$. Thus, at least $(nl \cdot (|E| + 1) - n)$ of the vertices $b_i'^{(u,v)}$ are in W. By a similar argument, if $b_i'^{(u,v)} \in W$, then at most n vertices of the form $c_{i,j}^e$ are not in W, and $(nl(|E| + 1) - n)|E| - n$ vertices of W are of the form $c_{i,j}^e$ for $j \in \{1, \ldots, |E|\}$. Let n_{C^e} be the number of vertices of W of color C^e. Thus, $n_{C^e} \geq (nl(|E| + 1) - n) \cdot |E| - n$. Let n_{blue} be the number of blue vertices in W. Then:

$$\begin{aligned} n_{blue} &\leq nl(|E| + 1) \cdot |E^W| + |E^W| \\ &\leq nl(|E| + 1) \cdot (|E| - 1) + (|E| - 1) \\ &= nl(|E| + 1) \cdot |E| - nl(|E| + 1) + |E| - 1 \\ &< nl(|E| + 1) \cdot |E| - n(|E| + 1) \\ &\leq n_{C^e} \end{aligned}$$

The strict inequality in the last but one step follows from the assumption that $l \geq 2$. Therefore, there are more vertices of color C^e than color blue in W. This contradicts the assumption that W is a winning district for blue. It follows that $E^W = E$ and V^W is a vertex cover.

It remains to prove that $|V^W| \leq l$. Suppose, for a contradiction, that $|V^W| \geq l + 1$. Then, the number n_{red} of red vertices in W is at least $n|E|(|E| + 1) \cdot (l + 1) - n$. On the other hand, there are in total $nl|E| \cdot (|E| + 1)$ blue vertices in H. Thus, we have $n_{red} \geq n|E|(|E| + 1)(l + 1) - n > nl|E| \cdot (|E| + 1) \geq n_{blue}$, a contradiction. Therefore, V^W is a connected vertex cover of size at most l. □

Corollary 3.4. *Gerrymandering is inapproximable in planar graphs, unless P = NP.*

Proof. Any approximation algorithm with a finite approximation guarantee must find at least one winning district if the maximum number of winning districts is strictly positive. By Theorem 3.3, this algorithm can then be used to distinguish between yes and no instances of the connected vertex cover problem in polynomial time. □

4 A CONSTANT FACTOR APPROXIMATION FOR PLANAR GRAPHS

Given our hardness bounds apply even in the setting of unweighted graphs, it is natural to instigate the study of approximation algorithms for planar graphs in the basic case of unweighted graphs. (We remark that unweighted graphs have been studied in their own right in [4, 20]). At first glance, Corollary 3.4 appears fatal to this endeavour: apparently, no approximation algorithms exist. However, this conclusion arises only because of the difficultly in distinguishing between cases where no districts can be won and cases where at least one district can be won. But this implies a large inapproximability bound only applies when the optimal number of winning districts w_{OPT} is small. Indeed our main result is the following.

Theorem 4.1. *There exists an algorithm with running time $O(n \log n)$ that computes a $(k, \lfloor w_{OPT}/O(c) \rfloor)$-partition in planar graphs.*

We remark that the floor function in Theorem 4.1 is necessary in view of Corollary 3.4. An important consequence of the theorem is that there is an $O(c)$-approximation algorithm for gerrymandering in unweighted planar graphs, provided the optimal number of winning districts w_{OPT} is a sufficiently large multiple of c. So for a constant number of candidates, there is a constant approximation algorithm given the optimal value is large enough.

We will now prove Theorem 4.1. Towards this aim, we first reduce our problem to the case where winning districts are singletons: given a (k, w_{OPT})-partition there is a feasible partition in which blue wins a large number of *singleton districts*. Let w_{OPT_S} denote the optimal solution where winning districts are singletons. Specifically, the next lemma implies that insisting all winning districts are *singleton* blue vertices will only incur a constant factor loss.

Lemma 4.2. *There is a (k, w_{OPT_S})-partition with singleton winning districts where $w_{OPT_S} \le \lfloor w_{OPT}/(2c+2) \rfloor$.*

Proof. Let G be a planar graph and consider its optimal (k, w_{OPT})-partition. Let W be the index set of the districts won by the blue candidate and let $\{D_i : i \in W\}$ be the corresponding set of winning districts in the (k, w_{OPT})-partition. Let $B_i \subseteq D_i$ and $B_W = \bigcup_{i \in W} B_i$ be the set of blue vertices in the winning districts. For each $i \in W$ we have $|B_i| \ge |D_i|/c$ because blue is the most preferred of the candidates. Hence $\frac{1}{c} \cdot \sum_{i \in W} |D_i| \le \sum_{i \in W} |B_i| = |B_W|$. We now create an auxiliary planar graph $\widetilde{G}$ as follows. For each losing district, contract the vertices in that district into a single vertex. This can be done as, by definition, each district induces a connected subgraph. Without loss of generality, we may color the resultant losing singleton vertices red. Since $\sum_{i \in W} |D_i| \le c \cdot |B_W|$, we have

$$\frac{w_{OPT}}{k} = \frac{\sum_{i \in W} 1}{|\widetilde{V}| - \sum_{i \in W}(|D_i| - 1)} \le \frac{\sum_{i \in W} |D_i|}{|\widetilde{V}|} \le \frac{c \cdot |B_W|}{|\widetilde{V}|} \quad (1)$$

As $\tilde{G}$ is connected it contains a spanning tree T. Let $deg_T(v)$ denote the degree of vertex v in the spanning tree T. It holds that:

$$\sum_{v \in B_W} deg_T(v) \le \sum_{v \in \widetilde{V}} deg_T(v) < 2|\widetilde{V}| \le \frac{2ck \cdot |B_W|}{w_{OPT}}$$

Here the final inequality follows from (1). Consequently, $\frac{1}{|B_W|} \cdot \sum_{v \in B_W} deg_T(v) < \frac{2ck}{w_{OPT}}$. Thus, the average degree in the tree T of the blue vertices in the winning district is less than $\frac{2ck}{w_{OPT}}$. Define $B'_W = \{v \in B_W : deg_T(v) \le (1 + \frac{1}{2c}) \cdot \frac{2ck}{w_{OPT}}\}$. By Markov's inequality, we have $|B'_W| \ge \frac{1+\frac{1}{2c}-1}{1+\frac{1}{2c}} \cdot |B_W| = \frac{|B_W|}{2c+1}$. Now take a maximal set $B^* \subseteq B'_W$ such that $\widetilde{G} \setminus B^*$ contains at most $k - |B^*|$ components. If $B^* = B'_W$ then selecting B^* as the winning singleton districts gives a factor $2c+1$ approximation guarantee.

Otherwise observe that removing any vertex $v \in B^*$ creates at most $\left\lfloor (1 + \frac{1}{2c}) \cdot \frac{2ck}{w_{OPT}} \right\rfloor = \left\lfloor \frac{k(2c+1)}{w_{OPT}} \right\rfloor$ new components (plus the singleton district v itself). Therefore, by the maximality of B^*, we have $(|B^*| + 1) \cdot \left(\left\lfloor \frac{k(2c+1)}{w_{OPT}} \right\rfloor + 1 \right) > k$. Rearranging, we have

$$|B^*| + 1 > \frac{k}{\left\lfloor \frac{k(2c+1)}{w_{OPT}} \right\rfloor + 1} \ge \frac{k}{\frac{k(2c+1)}{w_{OPT}} + 1} = \frac{w_{OPT}}{2c + 1 + \frac{w_{OPT}}{k}}$$

Because $|B^*| + 1$ is integral, the strict inequality yields $|B^*| \ge \left\lfloor \frac{w_{OPT}}{2c+1+\frac{w_{OPT}}{k}} \right\rfloor$. So the vertices of $|B^*|$ form the winning districts in a $(\hat{k}, \lfloor w_{OPT}/(2c+2) \rfloor)$-partition with $\hat{k} \le k$. We can convert the partition obtained from the previous algorithm into a $(k, \lfloor w_{OPT}/(2c+2) \rfloor)$-partition by splitting the losing districts. For each losing district that is not a singleton, we can construct a spanning tree on the districts and remove the leaves of the tree one by one. Each time we remove a leaf, we create a new losing district. We repeat this process until we have obtained k districts in total. This procedure is feasible because $k \le n$, and thus, we are guaranteed to obtain k districts before all the losing districts become singletons. □

Theorem 4.1 will follow immediately from Lemma 4.2 and the next lemma.

Lemma 4.3. *In planar graphs, there is an algorithm with running time $O(n \log n)$ that computes a $(k, \lfloor w_{OPT_S}/845) \rfloor)$-partition where the winning districts are singletons.*

So it suffices to prove Lemma 4.3 and we devote the remainder of this section to its proof. Through a more precise but slightly more intricate analysis of our algorithm, we could improve the constant by a factor of approximately 6. However, we opted not to pursue it, as the approximation constant would still be three digits. Finding a fast approximation algorithm with a small constant remains an open problem that we find interesting.

The restriction to the singleton winning districts case has some useful implications. In particular, we may assume without loss of generality that $c = 2$. This is because districts that are not blue singletons are considered losing no matter how many candidates there are. The following algorithm requires that we approximately know w_{OPT_S}, the maximum number winning singletons in a partition of G into k districts. Of course this number is unknown, but we can run the algorithm by guessing $\log n$ possible values of w_{OPT_S} (namely, powers of two) and selecting, from the different partitions obtained, the one that maximizes the number of winning districts. This trick costs us at most a factor 2 in the approximation guarantee.

Here is a quick overview of the approximation algorithm which consists of four phases. The algorithm aims to select the lowest degree blue vertex in a greedy manner and convert it into a winning singleton. However, to make this algorithm work effectively, we must first modify the graph. In the initial *pruning* step, we contract the red vertices that are connected and put them in the same losing district. Additionally, we convert blue vertices incident to many red leaves into red vertices since they are poor choices for a winning district. In the subsequent *cut-and-connect* step, we eliminate degree two and three red vertices and replace them with edges and triangles to maintain the graph structure. Although these vertices can significantly increase the degrees of blue vertices, they are not problematic since we can avoid making them losing singletons. Once these modifications are complete, we apply the *greedy* step to the resulting graph by selecting blue vertices with low degree to form singleton winning districts, as long as we do not generate more than k components. Finally, in the *5-color* step, the algorithm selects an independent set from the set of blue singleton winning vertices to ensure that the red vertices that were removed in the *cut-and-connect* step do not become singletons. Let's now detail these four phases.

Phase I: Pruning. The first step is the following pruning procedure which can be implemented in linear time.

Algorithm 1 Pruning

1: **Input:** a connected, planar graph G with vertices colored blue or red
2: Contract all connected red vertices
3: **while** $\exists$ a blue vertex incident to more than $\frac{12k}{w_{OPT_S}}$ red leaves (degree 1 red vertices in G) **do**
4: Color it red
5: Contract its red component into one red vertex
6: **end while**
7: **Output:** The resultant graph G_1

Lemma 4.4. *The graph G_1 output by the pruning procedure has the following properties:*

1. *The set R_1 of red vertices in G_1 forms an independent set.*
2. *Every blue vertex in G_1 is incident to less than $\frac{12k}{w_{OPT_S}}$ red leaves.*
3. *Any partition of G_1 into k districts induces a partition of G into k districts with the same number of winning blue singletons.*
4. *The set B_1 of the blue vertices of G_1 has cardinality $|B_1| \geq w_{OPT_S}/2$.*

Proof. Since two red vertices connected by an edge are contracted in G_1, we have that (1) holds. Property (2) holds due to line 3 of the algorithm. The only graph operation performed to obtain G_1 from G is contraction, hence, the connectivity of each district is maintained in G. So (3) holds. Next, we prove the fourth property. Let $G_{OPT} = (V_{OPT} = B_{OPT} \cup R_{OPT}, E_{OPT})$ be the graph obtained from the optimal decomposition of G into k districts, after the contraction of each losing district into a singleton red vertex. Observe that G_{OPT} is planar because it is a minor of G. Therefore, we have $\sum_{v \in V_{OPT}} deg_{G_{OPT}}(v) < 6 \cdot |V_{OPT}| = 6k$. It follows that, $\frac{1}{w_{OPT_S}} \cdot \sum_{v \in B_{G_{OPT}}} deg_{OPT}(v) < \frac{6k}{w_{OPT_S}}$. Therefore, the average degree of blue vertices in G_{OPT} is less than $\frac{6k}{w_{OPT_S}}$. Define $B'_{OPT} := \{v \in B_{OPT} : deg_{G_{OPT}}(v) \leq \frac{12k}{w_{OPT_S}}\}$. By Markov's inequality, we have $|B'_{OPT}| \geq \frac{1}{2} \cdot w_{OPT_S}$.

We will now prove that every vertex in B'_{OPT} remains blue upon termination of the pruning procedure (Algorithm 1). Observe that a blue vertex can only turn red (at line 4 in the algorithm) if it is incident to more than $\frac{12k}{w_{OPT_S}}$ red leaves at some point. For a contradiction, assume this is true for some vertex $v \in B'_{OPT}$. By definition of B'_{OPT}, we have $deg_{G_{OPT}}(v) \leq \frac{12k}{w_{OPT_S}}$. Therefore at the time when v is incident to more than $\frac{12k}{w_{OPT_S}}$ red leaves, at least two of those leaves, say r_1 and r_2 belong to the same losing district in OPT. Every district is connected so we can find a path P connecting those two leaves that does not pass through v. We have two case. First, at this time, all the vertices on P are red. But then r_1 and r_2 should have been contracted together, a contradiction. Second, there is a blue vertex b on P. But then r_1 and r_2 cannot be leaves at this time, a contradiction. Thus we have $|B_1| \geq |B'_{OPT}| \geq \frac{1}{2} \cdot w_{OPT_S}$. □

Phase II: Cut & Connect. The second step of the approximation algorithm is the cut and connect procedure. This procedure simply (i) replaces each red vertex of degree two by an edge connecting its two blue neighbours, and (ii) replaces each red vertex of degree three by a triangle connecting its three blue neighbours. This is illustrated in Figure 2.

Algorithm 2 *Cut and Connect*

Input: G_1 from Algorithm 1
Obtain a new graph G_2 from G_1 by turning all degree 2 and 3 red vertices into edges that connect the neighbouring blue vertices
Output: a new graph G_2

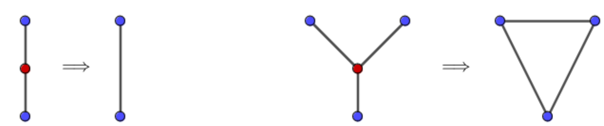

Figure 2: The Cut and Connect procedure

Lemma 4.5. *The graph G_2 output by the cut-and-connect procedure has the properties:*

1. *The set R_2 of red vertices in G_2 form an independent set.*
2. *Every blue vertex in G_2 is incident to less than $\frac{12k}{w_{OPT_S}}$ red leaves.*
3. *The set B_2 of blue vertices in G_2 has cardinality $|B_2| \geq w_{OPT_S}/2$.*
4. *G_2 is planar.*
5. *$|B_2| \geq \frac{|V_2|}{2+\frac{12k}{w_{OPT_S}}}$.*

Proof. The first three properties hold by Lemma 4.4. The fourth property holds because replacing each red vertex of degree two by an edge and replacing each red vertex of degree three by a triangle maintains planarity. So let's prove the fifth property that $|B_2| \geq \frac{|V_2|}{2+\frac{12k}{w_{OPT_S}}}$. To show this, we apply one last set of transformations. We create a new planar graph G_2^* from G_2 as follows:

- Delete all the red leaves of G_2.
- Delete all the edges between two blue vertices.

Clearly $G_2^* = (B_2 \cup R_2^*, E_2^*)$ is planar. Furthermore, it is bipartite because both the red vertices R_2^* and the blue vertices B_2 form independent sets. It is well-known that the average degree in a planar, bipartite graph is less than four. Moreover, because every red vertex in G_2^* has degree at least 4, the average degree of the blue vertices is less than 4. Hence:

$$4 \cdot |B_2| > \sum_{v \in B_2} deg_{G_2^*}(v) = \sum_{v \in R_2^*} deg(v) \geq 4 \cdot |R_2^*| \quad (2)$$

Here the equality holds because G_2^* is bipartite. Thus, from (2), we have $|B_2| > |R_2^*|$. Now consider G_2. Its vertex set is $V_2 = R_2^* \cup B_2 \cup \{\text{red leaves of } G_2\}$. Thus

$$|V_2| = |R_2^*| + |B_2| + |\{\text{red leaves of } G_2\}| \leq 2 \cdot |B_2| + \frac{12k}{w_{OPT_S}} \cdot |B_2| = \left(2 + \frac{12k}{w_{OPT_S}}\right) \cdot |B_2|$$

Here the inequality follows from (2) and the second property. Rearranging, we obtain $|B_2| \geq \frac{|V_2|}{2+\frac{12k}{w_{OPT_S}}}$. So the fifth property holds. □

Phase III: Greedy. The third step in the approximation algorithm is a greedy procedure.

Algorithm 3 *Greedy Algorithm on G_2*

Input: G_2 from Algorithm 2
while there exists a blue vertex that is not a singleton component **do**
  Pick the blue vertex v with the lowest degree
  if the graph obtained by removing the edges adjacent to v has less than k components **then**
    Remove the edges adjacent to v so that v is now a winning district
  end if
end while
Output: A partition $\{D_1, \ldots, D_{\hat{k}}\}$ of the graph G_2 where each D_i corresponds to a connected component and $\hat{k} \leq k$

Lemma 4.6. *A $(\hat{k}, \lfloor \frac{w_{OPT_S}}{169} \rfloor)$-partition of G_2 is output by the greeedy procedure, with $\hat{k} \leq k$.*

Proof. Again, because G_2 is planar, its average degree is less than 6. Furthermore, because $|B_2| \geq \frac{|V_2|}{2+\frac{12k}{w_{OPT_S}}}$, by property 5 in Lemma 4.5, the average degree of the blue vertices in G_2 is less than $6 \cdot (2 + \frac{12k}{w_{OPT_S}})$. Let $B_2' := \{v \in B_2 : deg_{G_2}(v) \leq 12 \cdot (2 + \frac{12k}{w_{OPT_S}})\}$. By Markov's inequality and the third property of Lemma 4.5, we have $|B_2'| \geq 1/2 \cdot |B_2| \geq 1/4 \cdot w_{OPT_S}$. Now take a maximal set $B^* \subseteq B_2'$ such that $G \setminus B^*$ (the graph obtained by deleting B^* from G) contains at most $k - |B^*|$ components. If $B^* = B_2'$ then selecting B^* as the winning singleton districts gives a factor 4 approximation guarantee. Otherwise, observe that any vertex $v \in B^*$ has degree at most $24 + \frac{144k}{w_{OPT_S}}$, hence removing v creates at most $25 + \frac{144k}{w_{OPT_S}}$ new components (including the singleton district v itself). By the maximality of B^*, adding one more element would increase the number of district above k. So the number of winning singletons is at least $\left\lfloor \frac{k}{25+\frac{144k}{w_{OPT_S}}} \right\rfloor \geq \left\lfloor \frac{k}{\frac{169k}{w_{OPT_S}}} \right\rfloor = \lfloor \frac{w_{OPT_S}}{169} \rfloor$, where the inequality comes from the fact that $w_{OPT_S} \leq k$. □

Phase IV: 5-Coloring. However, we are not done: we have a partition $\mathcal{D} = \{D_1, \ldots, D_k\}$ for G_2 but do not yet have a feasible partition for G_1 (and thus G). The issue is we deleted a collection of red vertices during the cut-and-connect procedure. We cannot add them to $\mathcal{D}$ as singleton districts as this will create too many districts. So we must connect these red vertices to existing districts in $\mathcal{D}$. But then the problem is these red vertices were only connected to blue vertices, so adding them to a winning singleton district will cause the district to be losing.

To overcome this observe that such red vertices had degree two or three and that, as shown in Figure 2, after deletion their neighbours become pairwise adjacent in G_2. Thus, if we choose a set of winning singleton districts that form an independent set then every deleted red vertex has at least one neighbour in G_2 that is in a losing district. We may then simply add the red vertex to that losing district. But can we quickly find a large set winning singleton districts that form an independent set? Yes, we simply take advantage of a significant result established in [8]: a planar graph can be 5-colored in linear time. This leads to the fourth and final step in our approximation algorithm, the following 5-color procedure.

Algorithm 4 *5-Color*

Input: G_2 and its partition from Algorithm 3
5-color the graph G_2
Take the vertices from the color class containing the largest number of winning singleton districts as the new winning districts
for each red vertex deleted in the Algorithm 2 **do**
  Add it to an adjacent losing district
end for
Output: a new partition $D_1', ..., D_{\tilde{k}}'$ of the graph G_1 with $\tilde{k} \leq k$

Proof. (of Lemma 4.3) Using the largest color class in the 5-coloring may cost an extra factor 5 in the approximation guarantee beyond that given by Lemma 4.6. Thus the final partition is a $(\tilde{k}, \lfloor \frac{w_{OPT_S}}{845} \rfloor)$-partition in G_1. By Lemma 4.4, it induces a $(\tilde{k}, \lfloor \frac{w_{OPT_S}}{845} \rfloor)$-partition in G. We can then turn it into a $(k, \lfloor \frac{w_{OPT_S}}{845} \rfloor)$-partition by splitting the losing districts as we did in the proof of Lemma 4.2. This concludes the proof of Lemma 4.3 and, thus, of Theorem 4.1. □

5 A PTAS FOR SINGLETON WINNING DISTRICTS

By Theorem 4.1, there is a constant factor approximation algorithm for the gerrymandering problem in planar graphs. This result holds for any number of candidates c, provided w_{OPT} is a sufficiently large multiple of c. But can we do any better? Is it possible that there is a polynomial time approximation scheme (PTAS) for the gerrymandering problem in planar graphs when w_{OPT} is a sufficiently large? This problem remains open. However, we can obtain a PTAS for a slight variant of the gerrymandering problem: the *singleton winning district gerrymandering problem.* This is the same as the standard gerrymandering problem except the objective is to maximize the number of *singleton* winning districts for the blue player. While this objective is contrived, from a theoretical perspective this is an important problem due to its close relation to the standard gerrymandering problem, as we illustrated by Lemma 4.2.

Theorem 5.1. *There is a polynomial-time approximation scheme (PTAS) for the singleton winning district gerrymandering problem in planar graphs.*

The first step in our PTAS is a preprocessing step based upon Baker's classical partitioning technique for planar graphs [3].

Algorithm 5 Preprocessing via Baker's Technique

Input: A planar graph G with vertices colored blue or red, an integer λ, and an integer $j \in \{0, \lambda - 1\}$
Fix a planar embedding P of G
Set $i = 1$
while P non-empty **do**
  Set L_i to be the vertices on the outer face of P
  if $i \equiv j \mod \lambda$ **then**
    color every vertex $u \in L_i$ red
    Contract all edges (u, v) with $u, v \in L_i$
  end if
  Delete L_i from P
  Set $i = i + 1$
end while
Output: The resultant graph $G_{\lambda,j}$

So, for a fixed λ, the preprocessing step creates a set of graphs $\{G_{\lambda,0}, G_{\lambda,1}, \ldots, G_{\lambda,\lambda-1}\}$. We claim that at least one of these graphs has a solution almost as large as the optimal solution in the original graph G. This is guaranteed by the next lemma.

Lemma 5.2. *With only singleton winning districts the following hold:*

(1) *Any (k, w)-partition of $G_{\lambda,j}$ induces a (k, w)-partition of G, for all k, w.*
(2) *For any λ, there exists $j \in \{0, \lambda - 1\}$ and $w^* \geq w_{OPT_S} \cdot \frac{\lambda-1}{\lambda}$ such that there exists a (k, w^*)-partition of $G_{\lambda,j}$.*

Proof. Observe that (1) holds trivially by construction of the preprocessing algorithm. So consider (2) Let λ be an integer and take a (k, w_{OPT_S})-partition of G for the singleton winning district gerrymandering problem. Denote by B_W the set of winning blue vertices. Then there exists j such that $|B_W \cap \bigcup_{i \equiv j \mod \lambda} L_i| \leq \frac{1}{\lambda} \cdot w_{OPT_S}$. Let $w^* := |B_W \setminus \bigcup_{i \equiv j \mod \lambda} L_i|$, the partition in G then induces a (k, w^*)-partition in $G_{\lambda,j}$, as required. □

Lemma 5.2 implies that by solving the singleton winning district gerrymandering problem in the appropriate graph $G_{\lambda,j}$ may produce a PTAS for the original graph G. But how can we solve the problem in $G_{\lambda,j}$. The key is to demonstrate that each graph $G_{\lambda,j}$ has bounded tree-width.

Lemma 5.3. *Each graph $G_{\lambda,j}$ has treewidth at most $3\lambda - 1$.*

Proof. The proof will use the that a λ-outerplanar has treewidth at most $3\lambda - 1$, as shown by Bodlaender [5]. Unfortunately we cannot apply Bodlaender's result directly to $G_{\lambda,j}$ because it may have an unbounded outerplanarity index. However, we can still utilize the result.

Specifically, we prove the lemma by induction on the size of $S = \{i : L_i \neq \emptyset, i \equiv j \mod \lambda\}$. For the base case, if $|S| = 0$, then the graph $G_{\lambda,j}$ has at most λ layers so its is λ-outerplanar and thus has treewidth at most $3\lambda - 1$.

For the induction step, consider $\bigcup_{i \leq j} L_i$ and $G_{\lambda,j} \setminus \bigcup_{i<j} L_i$. The former has treewidth at most $3\lambda - 1$ because it is λ-outerplanar. The latter has treewidth at most $3\lambda - 1$ by the induction hypothesis. To show that $G_{\lambda,j}$ has has treewidth at most $3\lambda - 1$, observe that the outer face of a connected planar graph is itself connected. Now consider a contraction step in the preprocessing algorithm. This implies that each vertex in L_j that remains after the contraction step must correspond to the outer face of a connected component in the graph $G \setminus \bigcup_{l<i} L_l$. Now let T be tree decomposition $\bigcup_{i \leq j} L_i$ after the vertices in L_j have been contracted. Further, for each vertex $x \in L_j$, let T_x be a tree decomposition of the corresponding connected component of $G_{\lambda,j} \setminus \bigcup_{i<j} L_i$. We may root each T_x at a node containing x in its bag. We can then merge T with T_x by connecting the root of T_x to any node of T that contains x in its bag. We do this for each $x \in L_j$. This gives a tree decomposition of width $3\lambda - 1$. To see that the resultant tree decomposition is valid observe that the paths in $G_{\lambda,j}$ connecting a vertex of T to a vertex of T_x must necessarily pass through the vertex x, which corresponds to the outer face of the connected component induced by the verticess in T_x. □

Proof of Theorem 5.1. Let $\epsilon > 0$. We choose $\lambda = \lceil \frac{1+\epsilon}{\epsilon} \rceil$. For each j, we run the dynamic programming algorithm described in Section 2 on $G_{\lambda,j}$. Since $G_{\lambda,j}$ has bounded treewidth by Lemma 5.3 and Theorem 2.1, this can be done in polynomial time. Now using Lemma 5.2, there exists a j such that the algorithm produces a $(k, w \cdot \frac{\lambda-1}{\lambda})$-partition of $G_{\lambda,j}$ inducing a $(k, w \cdot \frac{\lambda-1}{\lambda})$-partition of G. We picked λ such that $\frac{\lambda-1}{\lambda} \geq \frac{1}{1+\epsilon}$ so we get a $(1+\epsilon)$-approximation. Since this holds for any $\epsilon > 0$, we have a polynomial-time approximation scheme. □

We can combine Theorem 5.1 with Lemma 4.2 and get the following corollary.

Corollary 5.4. *For any $\epsilon > 0$, there exists an algorithm that can compute a $(k, \lfloor \frac{w_{OPT}}{(1+\epsilon)(2c+2)} \rfloor)$-partition for planar graphs in polynomial time.* [3]

We remark, that while the approximation constant obtained above is better than the one of Theorem 4.1, the running time is significantly worse.

6 CONCLUSION

Several interesting open problems remain. First, we proved that finding one winning district in a planar graph is NP-hard. But this reduction required a large number of candidates. If the number of candidates is a constant, is there a polytime algorithm to find one winning district? Second, we presented approximation algorithms for planar graphs, given w_{OPT} is sufficiently large. Our hardness result proves that this condition is necessary, but can the approximation guarantee be improved? Specifically, is there a polynomial time approximation scheme (PTAS) for the gerrymandering problem in planar graphs when w_{OPT} is sufficiently large? Third, as a special case, our approximation algorithms provided solutions where the winning districts are singletons. In most natural applications, the sizes of the districts are expected to be similar. Is there an approximation algorithm for planar graphs when the sizes of the districts are (roughly) equal?

[3]The precise running time is $O(n^{7+(1/\epsilon)(2c+2)})$.

REFERENCES

[1] M. Altman. 1997. The computational complexity of automated redistricting: Is automation the answer. *Rutgers Computer & Tech. LJ* 23 (1997), 81.

[2] N. Apollonio, R.I. Becker, F. Lari, I.and Ricca, and B. Simeone. 2009. Bicolored graph partitioning, or: gerrymandering at its worst. *Discrete Applied Mathematics* 157, 17 (2009), 3601–3614.

[3] B. Baker. 1994. Approximation Algorithms for NP-Complete Problems on Planar Graphs. *J. ACM* 41, 1 (1994), 153––180.

[4] M. Bentert, T. Koana, and R. Niedermeier. 2023. The Complexity of Gerrymandering over Graphs: Paths and Trees. *Discrete Appl. Math.* 324, C (2023), 103––112.

[5] H. Bodlaender. 1998. A partial k-arboretum of graphs with bounded treewidth. *Theoretical Computer Science* 209, 1 (1998), 1–45.

[6] N. Boehmer, T. Koana, and R. Niedermeier. 2023. A refined complexity analysis of fair districting over graphs. *Autonomous Agents And Multi-Agent Systems* 37, 13 (2023).

[7] A. Borodin, O. Lev, N. Shah, and T. Strangway. 2018. Big City vs. the Great Outdoors: Voter Distribution and How It Affects Gerrymandering.. In *IJCAI*. 98–104.

[8] N. Chiba, T. Nishizeki, and N. Saito. 1981. A linear 5-coloring algorithm of planar graphs. *Journal of Algorithms* 2, 4 (1981), 317–327.

[9] A. Cohen-Zemach, Y. Lewenberg, and J. Rosenschein. 2018. Gerrymandering Over Graphs. In *Proceedings of AAMAS'18*. 274–282.

[10] J. Dippel, M. D. la Tour, A. Niu, S. Roy, and A. Vetta. 2023. Gerrymandering Planar Graphs. *CoRR* abs/2312.14721 (2023).

[11] E. Eiben, F. V. Fomin, F. Panolan, and K. Simonov. 2020. Manipulating Districts to Win Elections: Fine-Grained Complexity. In *Proceedings of AAAI'20*. 1902–1909.

[12] R. Erikson. 1972. Malapportionment, gerrymandering, and party fortunes in congressional elections. *Am. Political Sci. Rev.* 4, 66 (1972), 1234–1245.

[13] B. Fleiner, B. Nagy, and A. Tasnádi. 2017. Optimal partisan districting on planar geographies. *Cent. Eur. J. Oper. Res.* 25, 4 (2017), 879–888.

[14] A. Fraser, B. Lavallee, and B. Sullivan. 2022. Gerrymandering Trees: Parameterized Hardness. *ArXiv* abs/2205.06857 (2022).

[15] M. Garey and D. Johnson. 1977. The Rectilinear Steiner Tree Problem is NP-Complete. *SIAM J. Appl. Math.* 32, 4 (1977), 826–834.

[16] S. Gupta, P. Jain, F. Panolan, S. Roy, and S. Saurabh. 2021. Gerrymandering on Graphs: Computational Complexity and Parameterized Algorithms. In *Proceedings of SAGT'21*. 140–155.

[17] J. Håstad. 1999. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica* 182, 1 (1999), 105 – 142.

[18] S. W. Hess, JB Weaver, HJ Siegfeldt, JN Whelan, and PA Zitlau. 1965. Nonpartisan political redistricting by computer. *Operations Research* 13, 6 (1965), 998–1006.

[19] S. Issacharoff. 2002. Gerrymandering and political cartels. *Harvard Law Review* (2002), 593–648.

[20] T. Ito, N. Kamiyama, Y. Kobayashi, and Y. Okamoto. 2019. Algorithms for Gerrymandering over Graphs. In *Proceedings of AAMAS'19*. 1413–1421.

[21] Y. Lewenberg, O. Lev, and J. S. Rosenschein. 2017. Divide and conquer: Using geographic manipulation to win district-based elections.. In *Proceedings of AAMAS'17*. 624–632.

[22] D. Lubin. 1999. *The paradox of representation: Racial gerrymandering and minority interests in Congress*. Princeton University Press, New Jersey.

[23] C. Puppe and A. Tasnádi. 2009. Optimal redistricting under geographical constraints: Why "pack and crack" does not work. *Econ. Lett.* 105, 1 (2009), 93 – 96.

[24] A. Stoica, A. Chakraborty, A. Dey, and K. Gummadi. 2020. Minimizing margin of victory for fair political and educational districting. In *Proceedings of AAMAS'20*. 1305–1313.

[25] A. Trelles, M. Altman, E. Magar, and M. McDonald. 2014. The Effects of Automated Redistricting and Partisan Strategic Interaction on Representation: The Case of Mexico. *Available at SSRN 2486885* (2014).

[26] M. Wines. 2019. What Is Gerrymandering? And How Does it Work? *New York Times* (June 2019).

[27] D. Zuckerman. 2006. Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number. In *Proceedings of STOC'06*. 681–690.