Little House (Seat) on the Prairie: Compactness, Gerrymandering, and Population Distribution

Allan Borodin[1], Omer Lev[2], Nisarg Shah[1], and Tyrone Strangway[2]

[1]University of Toronto, {bor,nisarg}@cs.toronto.edu
[2]Ben-Gurion University of the Negev, {omerlev,styrone}@bgu.ac.il

Abstract

Gerrymandering is the process of creating electoral districts for partisan advantage, allowing a party to win more seats than what is reasonable for their vote. While research on gerrymandering has recently grown, many issues are still not fully understood such as what influences the degree to which a party can gerrymander and what techniques can be used to counter it. One commonly suggested (and, in some US states, mandated) requirement is that districts be "geographically compact". However, there are many competing compactness definitions and the impact of compactness on the gerrymandering power of the parties is not well understood.

We develop a modular, scalable, and efficient algorithm to automatically draw districts by optimizing a variety of criteria. We confirm its effectiveness on several US states by pitting it against maps "hand-drawn" by political experts. We then use it to study the interaction between population distribution, geographic compactness constraints, and political power. Specifically, we conduct experiments using real data from US political elections and the Polsby-Popper score, a widely accepted compactness definition. Our experiments show that imposing compactness constraints does limit the power to gerrymander, but only in ruling out extreme gerrymandering possibilities. Furthermore, regardless of how strict the constraint, the more-rural party maintains greater gerrymandering power than the more-urban party.

1 Introduction

In many democracies, politicians are elected to represent the people of particular geographic areas, called districts[1]. There is no global, country-wide, vote, but instead voters within a district pick a winner from the alternatives vying to represent their district. The overall winner is the alternative which won the majority (or sometimes a plurality) of districts. This method of decision-making by using bottom-up structures is not unique to countries, of course, and can be seen in organizations (e.g., universities reaching decisions by approving them at the departmental level, and if enough departments support, at the Faculty level, etc.), and other structures where sub-unit level divisions make sense.

How voters are partitioned into these districts directly affects the makeup of the legislative body. The partitioning is often governed by hard constraints. For example, most jurisdictions require that the districts be geographically connected (with certain exceptions) and have roughly equal numbers of voters. In addition, there are many competing goals when designing a districting plan [Wasserman, 2018]. One could

[1]Many countries use more specific terminology for these, such as constituencies, ridings, and others. We shall use the term 'district' to refer to them in general.

prioritize not breaking up communities of interest, such as those with a shared culture and history.[2] It may also be desirable to be as compatible as possible with established city and county boundaries, a consideration studied by Wheeler and Klein [2020]. Another reasonable goal would be to obtain geographically compact regions (a goal enshrined in some US states' laws and regulations[3]). A less defensible, but fairly common, goal is *gerrymandering*: designing districts for partisan gain, i.e., creating districts which help a particular party gain a number of seats beyond its popular support.

In the US, following every 10-year census, state legislatures decide their new federal congressional and state legislative districts, and partisan concerns are often part of the consideration [Wang, 2016]. For example, in the 2020 federal election in North Carolina, a state accused of gerrymandering (partially overturned by courts [Common Cause v. Lewis, 2019]), the Democratic party received 49.96% of the vote and won five districts; the Republican party received 49.41% of the vote and won eight districts. As noted above, due to the many competing goals, even with clearly stated goals it is not clear what is fair or optimal when it comes to non-partisan districting (see Wasserman [2018] for further discussion and a comparison of objectives).

Parallel to the political partisan redistricting process there is a more complex, ongoing population-wide process. In the US [Bishop, 2009; Rodden, 2019] and Europe [Noack, 2016], voters are reorganizing themselves for various economic and social reasons. The consequence is that voters of left leaning parties tend to cluster in urban centres; while voters of right leaning parties tend to spread out in rural regions.

Our contributions. Our work explores aspects of both of these processes – both the immediate partisan one, as well as the process of population dynamics. In the first part of our work (Section 5), we introduce our automated redistricting procedure, which, unlike some previous work, operates on the scale of real-world data. Our method is flexible and can be used to design plans for various objectives, both partisan and nonpartisan. Compared to districts drawn by political experts, we almost always match, and even sometimes exceed, their performance.

Once we show the power of our algorithm, we begin using it to understand the interplay between population distribution, geographic compactness constraints, and political power. Our algorithm allows us to examine possible requirements that have been suggested as a means to mitigate or eliminate gerrymandering. In particular, we study the impact of a compactness requirement. In Section 6, a few compactness measures are considered and we see that in the US, the more rural party (Republicans) still consistently outperforms the more urban party (Democrats). Moreover, we see that this advantage is robust *even in a non-gerrymandered, optimally-compact plan*. This advantage is necessarily not due to political gaming of the division process, but rather due to the geographic spread of each party's supporters.

We further examine, in Section 7, how much compactness constraints affect gerrymandering possibilities. We see that, indeed, demanding stringent compactness constraints reduces the ability of parties to gerrymander. However, in most cases, the compactness requirement allows for relatively greater rural-party gerrymandering. Moreover, under the most stringent compactness constraints, the urban party sometimes cannot even optimally achieve its vote proportion.

2 Related Work

Political districting, and in particular gerrymandering, has long been studied within various fields. There have been legal discussions [Schuck, 1987; Issacharoff, 2002; Friedman and Holden, 2009], and recently the US Supreme Court ruled [Rucho v. Common Cause, 2019] that partisan gerrymandering cannot be ad-

[2] At least regarding ethnic minorities, this is required by the US' Voting Rights Act of 1965.

[3] For example, California's constitution states, in article XXI, "districts shall be drawn to encourage geographical compactness".

dressed by federal courts. Gerrymandering has long been studied from a historical perspective [Engstrom, 2006; Butler, 1992] and by political scientists [Erikson, 1972a; Grofman *et al.*, 1997; Tangian, 2010; Felsenthal and Miller, 2015].

Automated redistricting for various objectives has been widely studied in the computer science and mathematics literature, since Vickrey [1961] suggested its possibility (see Becker and Solomon [2021] for a recent summary). Various methods have been suggested for detecting gerrymandering [Wang, 2016; Grofman and King, 2007; Puppe and Tasnádi, 2008; Fifield *et al.*, 2018]. Another common goal is to accept that parties will gerrymander when possible, and design systems that reduce the possible gain due to gerrymandering. One such suggestion, which we explore in this work, is to enforce a geographic compactness constraint on the newly drawn districts. A different suggestion due to Pegden *et al.* [2017] is to allow all parties to participate in the districting process. They design an algorithm, inspired by the "cut and choose" procedure for cake-cutting, such that when all parties simultaneously participate to optimize their own gain, no party enjoys too strong an advantage. It is also worth noting that finding an optimally gerrymandered solution is known to be NP-hard in various settings [Lewenberg *et al.*, 2017; Dyer and Frieze, 1985; van Bevern *et al.*, 2015].

The AI community has been exploring dividing voters into groups [Bachrach *et al.*, 2016; Borodin *et al.*, 2019; Lev and Lewenberg, 2019] and, more concretely, geographic concerns, as in cake-cutting [Segal-Halevi *et al.*, 2020]. This community has explored gerrymandering and its connection to voter distribution. Cohen-Zemach *et al.* [2018] examined gerrymandering in a random graph model and Lewenberg *et al.* [2017] examined gerrymandering when different parties have support in different urban centres, both finding that greedy algorithms work decently well on their simulated data. In the latter work, each party has a different urban strength, unlike our work (and many real-world voting patterns), where there are distinct urban and rural parties. The methods suggested in the latter two papers ignore (or severely stretch) population bounds (i.e., the constraint that the districts have roughly equal numbers of voters), creating districts that would be *highly* illegal in most jurisdictions. Closer to our work, Borodin *et al.* [2018] examined gerrymandering in urban-rural settings. They worked on small and structured simulations (16×16 grid graphs), and used an integer linear programming based approach to find optimally gerrymandered solutions that respect population bounds. They found that the rural party has a significant advantage over the urban party in terms of gerrymandering. However, their method is unable to scale beyond their small and structured examples. In contrast, our automated procedure, based on simulated annealing, is able to scale to real-world graphs with tens of thousands of nodes while still respecting population bounds. Also, to the best of our knowledge, ours is the first work in the AI community which explores the role of geographic compactness in gerrymandering.

3 Model

We examine gerrymandering with a graph-theoretic formulation. We shall be using a US-oriented terminology (states, precincts, etc.), but the formulation represents most geographic districting settings. A state is an undirected graph $G(V, E)$. Each node $v \in V$ represents a precinct, a small geographic region where votes are tallied.[4] An edge $(u, v) \in E$ represents that precincts u and v share a physical boundary. For $v \in V$ let n_v^e be the number of people who live in precinct v and vote in election e, and $n_v^{p,e}$ be the number of people who live in v and vote for party p in election e. We will omit e when the context is obvious. Let

[4] In the US, census block data is more fine grained, but it does not contain any voting information, so it is not useful for our needs.

$N = \sum_{v \in V} n_v$ be the total number of voters in the state. We limit our focus to two parties: the rural party (in the US, Republicans (R)), and the urban party (in the US, Democrats (D)).

Creating a districting plan requires partitioning G into K vertex-disjoint subgraphs $G_1, \cdots, G_K$ (the districts). The number of districts K is extrinsically determined (in the US, by a census every 10 years). There are two widely accepted requirements for legal districts in the US and elsewhere:

Contiguity For each $k \in [K]$, G_k must form a connected subgraph of G. In the real world, this translates to being able to walk from any point in the district to any other point in the district without crossing into another district.

Population balance-δ Given $\delta > 0$, for each $k \in [K]$,

$$1 - \delta \leq \frac{\sum_{v \in V(G_k)} n_v}{N/K} \leq 1 + \delta.$$

The exact value of δ required in the U.S. changes between states (and judicial decisions). Informally, the criteria is that districts should be as near equal-sized in population as possible [Levitt, 2020]. We take $\delta = 0.005$, so that the maximum population deviation between any two districts is at most 1% of the average population per district. This is the legal requirements in some states, and a far tighter constraint than what is respected by previously proposed automated redistricting methods which can scale to real data.

Parties are interested in winning as many districts as they can. The party with the most voters in a district is typically said to win that district. For example, if $\sum_{v \in V(G_k)} n_v^{D,2012} > \sum_{v \in V(G_k)} n_v^{R,2012}$, we say the Democrats win district k according to the 2012 presidential vote totals in a given state. If the inequality is reversed, we say the Republicans win the district in that election. Ties are broken arbitrarily, though they have not occurred in our experiments. Note that this is only one way to define "winning a district". In Section 5.1 we will look at winning as a probabilistic event, blending vote totals from prior elections.

4 Election Settings

In this paper, we primarily use election data from three US states — Pennsylvania, North Carolina, and Wisconsin — from the 2012 and 2016 presidential elections. These are the elections for which granular, precinct-level, data is available.[5] Each of these three states has a particular election of interest. For the reference of scale, we also include the number of nodes (precincts) and edges in the graphs of each state. In comparison, prior automated redistricting algorithms which respect the tight population balance constraints stated above worked only for 16×16 grid graphs (256 nodes, 480 edges).

Pennsylvania (PA) 2012 *Sizeable Democrat advantage*. The Democratic candidate (Barack Obama) won 51.97% of the vote, vs. 46.59% to the Republican candidate (Mitt Romney). PA has 9,255 nodes and 25,721 edges.

North Carolina (NC) 2016 *Sizeable Republican advantage*. The Republican candidate (Donald Trump) won 49.83% of the vote, vs. 46.17% to the Democratic candidate (Hillary Clinton). NC has 2,692 nodes and 7,593 edges.

Wisconsin (WI) 2016 *Near tie*. The Republican candidate (Donald Trump) won 47.22% of the vote, vs. 46.45% to the Democratic candidate (Hillary Clinton). WI has 6,634 nodes and 18,126 edges.

[5] Data from MGGG (https://github.com/mggg-states).

In addition to these particular vote outcomes, they also provide a good mix of geographic features. WI, for example, has its north-east corner carved up by lake Michigan, forming a jagged bay. PA and NC, on the other hand, have a much more convex shape. Furthermore, the population distribution is varied: PA's large urban centres are in its east and west edges, whereas in NC, the urban centres are concentrated in the middle of the state.

4.1 Voting and the Urban-Rural Divide

State	2012 correlation	2016 correlation
NC	0.79	0.80
PA	0.47	0.58
WI	0.38	0.57

Table 1: Spearman correlation between a precinct's fraction of D party votes and its density (total population divided by area) in three states and two elections.

As noted by political scientists [Bishop, 2009; Rodden, 2019] and political commentators [MacGillis, 2016], the key geographic feature of US political parties is a growing divide between the more rural Republican party and the more urban Democratic party. We follow elections in three widely different states, with differing ethnic makeup, education patterns, and history. However, this feature is common across all three states in our data as well: densely populated urban centres favour the Democrats, while sparsely populated rural regions favour the Republicans (see Table 1).

5 The GREAT Algorithm

Before we can study the role of compactness and population distribution in gerrymandering, we need an algorithm that can optimize for various compactness and partisan fairness metrics (or handle them as constraints) on real-world data which has interesting population distribution patterns embedded in it. To that end, we introduce our Goal-based Redistricting for Elections Automatically using Technology (GREAT) algorithm, an algorithm that can create districts from graph representations. As we demonstrate later in the paper, our algorithm, with minimal engineering effort, can be used to optimize various measures of partisan fairness (e.g., to minimize the robust partisan bias metric introduced in Section 6), partisan gain (e.g., the number of districts won by a given party either by achieving a plurality of votes, or at least a threshold fraction of votes, or with at least a certain probability), and compactness (e.g., the Polsby-Popper and Convex Hull scores defined in Section 6). Furthermore, the algorithm can optimize towards one of these goals while satisfying strict constraints on other metrics (e.g., optimize compactness while ensuring that a given party wins at least a fixed number of districts). In Section 5.1 we show our algorithm is capable of creating highly partisan districtings, comparable to state of the art hand-crafted ones. In Section 6.1, we show it is capable of generating districtings that are far more compact than those used in practice. First, we give a brief overview of our algorithm (see Appendix A.1 for a full description).

Our method is based on simulated annealing (SA), a local-search like method which occasionally makes non-improvement steps, allowing it to escape local optima. After some fixed number of iterations or elapsed time, the process is terminated and the best of all iterated solutions is returned. Starting from a (often random) initial districting, a step is considered by using a modification of the tree-recombination procedure proposed by the Metric Geometry and Gerrymandering Group [MGGG, 2020]. Briefly, the method takes

State	Total seats	Our D	538 D	Our R	538 R
MD	8	7	5 (8)	4	4 (4)
MA	9	9	9 (9)	0	0 (0)
NC	13	7	8 (8)	11	10 (10)
PA	18	8	8 (9)	13	13 (13)
WI	8	5	5 (5)	6	6 (6)

Table 2: The first column is the total number of seats in the state. The second and third columns are the number of districts D take with over 82% probability with our algorithm and the 538 optimally-gerrymandered districting without compactness restrictions, respectively. The fourth and fifth columns are the same for the R party. The 538 numbers show the number of districts won according to their districting based on our election data. In parentheses are 538's reported results using absentee data to which we did not have access to.

a set of m adjacent districts, from the current solution, and recombines and redivides the nodes in them to form m new districts. This is done by drawing random spanning trees over the precincts of the m districts and cutting random edges in the trees to separate the nodes into the desired number of districts. For performance reasons, we generally use $m = 2$. Using $m = 3$ and $m = 4$ did not result in any improvement in the quality of the results, but led to significantly increased runtimes. In short, any objective that can be expressed numerically and calculated from an arbitrary districting may be used. Additionally, any binary constraint (for which it can be efficiently checked whether a given districting satisfies it) can be incorporated by ensuring that the algorithm only considers steps which satisfy this constraint.

5.1 Proof of Concept: Gerrymandering Districts

Unfortunately, like the body of work before us, we are unable to provide guarantees (e.g., convergence to optimal or approximately optimal solutions) on our method's performance. Instead, we will compare against Nate Silver's 538 gerrymandering project [Silver, 2020]. The election experts at 538 hand-crafted thousands of electoral districts for various objectives. While there is no guarantee their districtings are optimal, there are no publicly available ones which surpass them.

As noted above, winning a plurality of votes is just one of the measures of what it means to win a district. At 538, they took a probabilistic view, designing partisan districtings that maximized the number of districts that were won with a sufficient probability. Unfortunately, they released few details regarding their method. But, we believe we were able to reconstruct it using released results (see Appendix A.2 for a detailed description of our reconstruction). Briefly, 538 uses the Cook Partisan Voting Index (CPVI) [Cook, 2018], which measures a district's D party bias according to the 2012 and 2016 elections, and transforms it into the probability that the D party wins that district. The R party wins it with the remaining probability. When gerrymandering for party P, 538's objective was to maximize the number of districts for which P's probability of winning was at least 82%. To guide our method, we used a combination of the expected number of districts won by P and the total number of districts won by P with at least 82% probability. For an exact breakdown of this technique, see Appendix A.3.

The availability of presidential election data at the precinct level is inconsistent. Thus, we are unable to compare the districtings our method generated against 538's on all possible states. There are five states for which we have publicly available data, and for each of them we optimized for the 538 objective for each party. For each state and party we ran our algorithm on 60 cores with 2.10 GHz computing power for 24 hours, though we found the algorithm often stopped advancing well before this deadline. Out of all solutions iterated in all threads, we took the one with the most districts above 82% for our target party. Our results are shown in Table 2. We note the values 538 reported on their website include absentee data (mail-in ballots), which is not publicly available at the precinct level for us, so we do not use it when comparing districtings.

Overall, there was only one case, NC for D, where we did not match the 538 value. Even here, we only missed by one district out of the 13. We did outperform 538 in Maryland for the Ds, but we caution each party had over 25% of their vote come from absentee ballots[6]. In NC for the R party, we also outperformed 538.

5.2 The Ethics of Automated Redistricting

Before discussing our main results, we wish to touch upon the ethics of automated redistricting and its potential implications. There is an understandable concern such a tool could be used to negatively impact democratic freedoms. This point is especially salient for our tool, which, in hours, can match what human experts take much longer to produce.

We believe the marginal utility of such a tool, for gerrymanderers and others who seek to disenfranchise voters, is minimal. The redistricting process takes years, and is only actually done once every ten years in the United States (and many other democracies). In these situations, nefarious parties would have years, and often near unlimited resources, to have experts hand design districts. And as we have shown, they still are able to occasionally outperform our automated process.

Instead, we see the real use of our tool as something researchers can use. As we will show, our tool helps illustrate the geographic bias in single winner electoral districts. With it, we will be able to calculate the effectiveness of proposed solutions for gerrymandering such as compactness requirements. Furthermore, it can be used to help combat gerrymandering. If our algorithm produces districtings that are as partisan as ones that have been implemented, this can be seen as evidence that the real districts are gerrymandered. Because our tool is highly modular, it can be used to quickly propose alternative districtings. These may not be the final ones implemented, but they can provide an idea of what is possible.

6 Designing Compact Districts

As mentioned, compactness is often a legislated requirement, even if the mathematical definition and the required level of compactness are not specified. Furthermore, it is not clear if compact plans are more free of partisan bias than less compact ones. Thus, in this section we study plans designed to optimize various notions of compactness. We examine these plans from a partisan perspective and contrast them with currently used plans. We find that various definitions of compactness can reduce partisan bias, relative to plans used in real life. Despite this, we find a persistent partisan bias to these compact plans, i.e., they favour the R party despite being optimized for a non-partisan goal.

To measure compactness of a district within a plan we use the following measures (formal definitions in Appendix A.4):

Polsby-Popper (PP) The Polsby-Popper score of a district is the ratio of the district's area to the area of the circle with the same perimeter as the district.

Convex Hull (CH) The Convex Hull score of a district is the ratio of the district's area to the area of the minimum convex hull that bounds the district.

[6] In the other 4 states at most 0.3% came from absentee ballots.

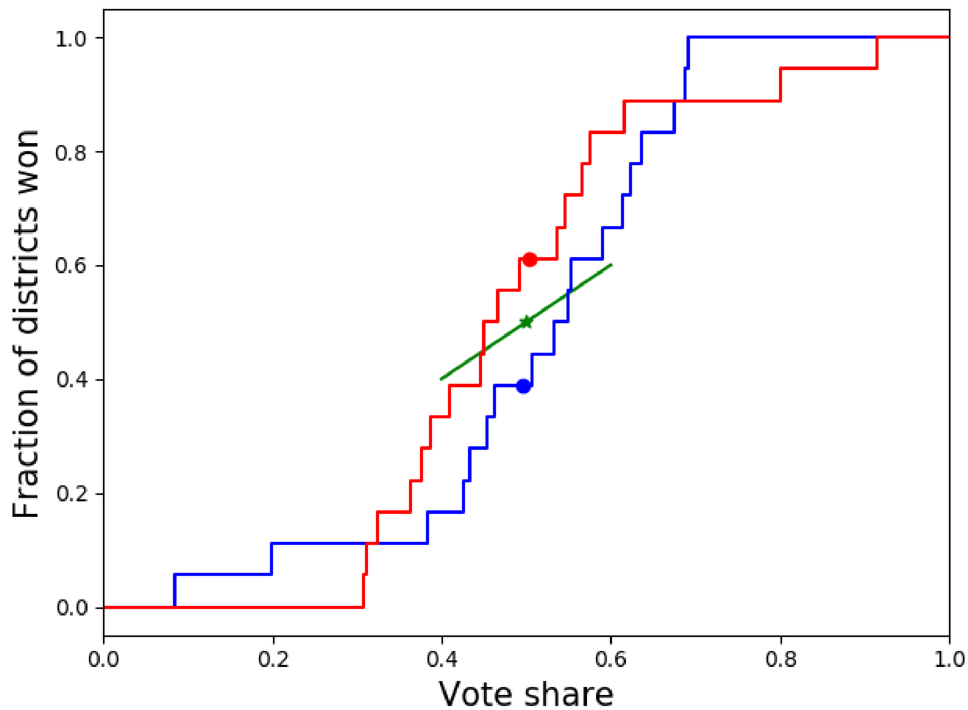

Figure 1: Uniform swing for each party (R in red, D in blue) in the 2016 presidential election in PA under our Polsby-Popper compact plan. Vertical axis shows the fraction of districts won; horizontal axis the vote fraction. The dots on the party curves indicate the actual election outcome (0 swing). The green line is the range of proportional outcomes on the range $[0.4, 0.6]$. The point $(1/2, 1/2)$ is marked with the green star.

For both these metrics, we use the GREAT algorithm to find the plan optimizing the average of the compactness measure across all the districts in the districting. Because each districting optimizes for a different compactness goal, and there is no universally-agreed-upon definition of compactness, it is difficult to compare them in terms of their compactness. However, we can measure whether these plans provide partisan fairness.

As mentioned above, we use proportionality as our notion of partisan fairness. Formally, we want the fraction of districts won by each party to match as closely as possible to its fraction of votes statewide. However, measuring how proportional a districting is based on the data from just one election is not very robust. To address this, the existing work uses the so-called *uniform swing* model [Erikson, 1972b]. In this model, hypothetical elections are generated starting from a baseline election by shifting the vote shares of the parties. Specifically, the vote share of a given party is increased or decreased by an equal amount in every district (capping the vote shares by 0% from below and 100% from above). The fraction of districts won by each party is then measured in these hypothetical elections in order to measure how much proportionality would be violated if the vote shares change over time. See Figure 1 for the uniform swing model applied to the 2016 PA presidential election under the plan produced by our algorithm for optimizing the Polsby-Popper score, where the fraction of districts won by a party is plotted against the fraction of votes received by the party at different uniform swing levels.

There are several metrics that use the uniform swing model to measure the partisan bias in a given districting. We are interested in the *partisan bias* score [Grofman, 1983]. This value measures the vertical

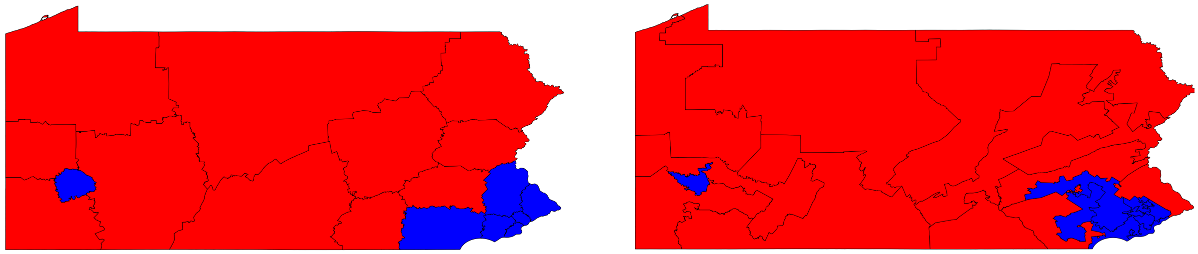

Figure 2: PA districts (R wins in red); (D wins in blue) based on the 2016 PA election data. The upper one is our plan, optimizing the convex hull score; the lower one is PA's actual 2011 districts.

displacement of the swing curve from the point $(1/2, 1/2)$.[7] Intuitively, the partisan bias measures the divergence from the idea that "half the votes should translate to half the seats". More generally, we can measure the vertical displacement from any point (a, a) for $a \in [0, 1]$ ("an a fraction of the vote share should translate into an a fraction of the seats").

We introduce a robust version of this metric. Fixing a line segment $[l, r]$ ($l, r \in [0, 1], l < r$), we measure the average vertical distance (signed) from the swing curve to the line $y = x$ over this line segment. That is, we measure what fraction of districts, on average, a party wins over a proportional division in a reasonable range of vote shares. We use $[0.4, 0.6]$ as the reasonable range (i.e. when the vote shares of both parties are between 40% and 60%).[8] The $45°$ line in this range is shown in green in Figure 1. Like before, note that the fraction of districts won by one party with $0.5 - t$ vote share is exactly one minus the fraction of districts won by the other party with $0.5 + t$ vote share, for any t. Hence, as long as the range is symmetric around the 0.5 vote share point, the average vertical distance of both parties' swing curves to the green line will be identical in value (but opposite in sign), and either can be measured. Intuitively, the larger the absolute value of this distance, the greater the bias the plan has towards one of the political parties (which one depends on the sign).

6.1 How Compact is Compact?

Our algorithm is capable of designing highly compact, and legal, plans. As mentioned above, there is no universally accepted notion of compactness, so we optimized for two popular notions: the Polsby-Popper (PP) score and the Convex Hull (CH) score. Unfortunately, there are no publicly available districtings that optimize for these measures for us to compare against as benchmark. From a visual standpoint (see Figure 2), our plans pass an "eye test" for looking compact, especially compared to the plans enacted in practice. While each plan was optimized for its own metric, there was a fair bit of correlation – a plan optimized for one measure of compactness was usually also quite compact according to the other measure.

6.2 Compactness Can Improve Fairness

In our experiments, we find that optimizing for compactness typically yields districtings that have improved partisan fairness relative to the districtings enacted in 2011, according to our robust partisan bias score. This improvement is consistent across several states and independent of the compactness measure optimized. Figure 3 shows our robust partisan bias score (the lesser the fairer) for various districtings in three states using the 2012 presidential election data. The purple bars represent the 2011 districtings, the blue and green

[7]We can take the distance of the swing curve of either party as both distances are guaranteed to be identical.

[8]The popular votes in most presidential elections fall in this range.

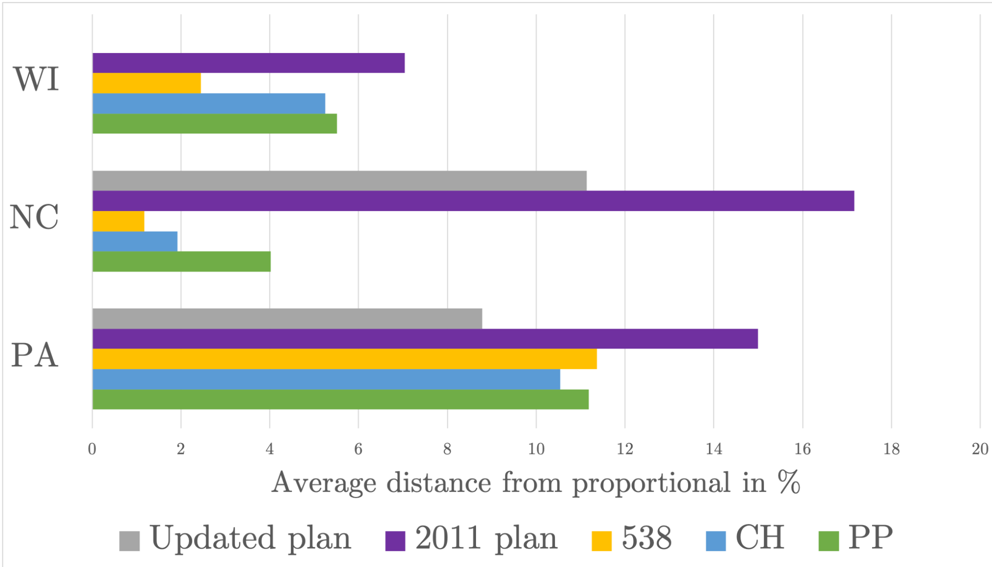

Figure 3: Average distance from the R swing curve to the $y = x$ line over the range $[40, 60]$ for the different compactness criteria in each state using 2012 presidential election data. The WI 2011 plan was not struck down, unlike in PA and NC, thus there is no "new" plan for it. See Appendix A.4 for additional swing curves.

bars indicate districtings produced by our algorithm for optimizing the CH and PP measures, respectively, and the yellow bars indicate the districtings produced by 538 for optimizing a different compactness measure (the average distance of a resident in the district from the centroid of the district). Note that all compact plans exhibit a lower bias towards the R party than the 2011 plans do. This pattern holds also when using the 2016 presidential election data.

This improvement is sometimes extreme: in NC, the 2011 districting (with a 17% robust partisan bias towards the R party) is more than three times as biased as any of the compact districtings. It is worth noting that both NC and PA 2011 districtings were struck down by the courts for being overly biased. The NC 2011 districting was found to disenfranchise minority voters [Common Cause v. Lewis, 2019] while in PA the plan was found to disenfranchise Ds [LWV v. Pennsylvania, 2018]. The updated districting from 2016 (with an 11% robust partisan bias towards the R party) was still more than twice as biased than any compact districting. We also remark that the results remain qualitatively similar even if we use a different interval for reasonable vote range in our definition of robust partisan bias.

Interestingly, the updated districtings from 2016 in NC and PA seem quite dissimilar. While the updated NC districting is still significantly more R-biased than any of the compact ones, the exact opposite holds for the new PA districting. Its level of robust partisan bias is slightly lower than the compact plans, although it is, of course, less compact. This may have to do with the new PA plan possibly being designed with partisan proportionality in mind [Cohn *et al.*, 2018]. That said, it should be noted that neither of the compact plans are designed to optimize the robust partisan bias metric. In fact, when we use our algorithm to optimize for this metric specifically, we can find districtings that have near-zero robust partisan bias in all three states (these are not shown in the figure because the bars would be virtually invisible).

6.3 The Rural Advantage

For all plans, in all three states, both elections, and both ranges of comparison there is one consistent pattern: The R party always has a positive score in our metric[9], that is, the more rural party can expect to gain more seats than its proportional voter share. This is despite the fact every single one of these plans was designed only to optimize some notion of compactness (one was even supposedly designed to maximize proportional fairness — giving each party a fraction of seats comparable to their fraction of votes statewide). This advantage can be significant, with the compact plans in PA this was a 10% (and often higher) advantage in seat share on average.

7 Designing Partisan Districts

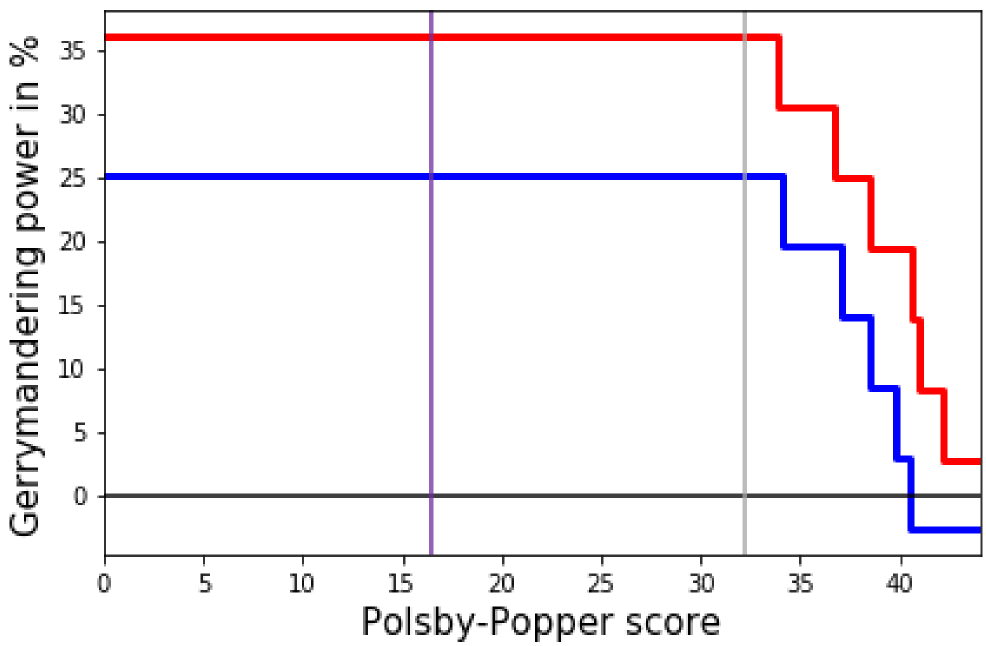

Figure 4: Gerrymandering power when faced with a minimum required Polsby-Popper score using data from the 2016 PA presidential election. R in red; D in blue. The vertical purple line is the Polsby-Popper score of the 2011 congressional plan, the vertical grey line is the Polsby-Popper score of the 2018 court mandated plan. Average distance between the two curves is a 10.8% advantage for the Rs. The Polsby-Popper scores for Wisconsin and North Carolina are given in Appendix A.5.1.

In this section we examine what limits do fixed compactness thresholds impose on partisan gerrymandering. As we saw in the previous section, compactness constraints can lead to more balanced outcomes, but we do not know what restrictions are necessary. For compactness, we use the average Polsby-Popper score of a plan. To measure gerrymandering ability, we use *gerrymandering power* (introduced in Borodin *et al.* [2018]). For a particular election, the gerrymandering power of party p is defined as the difference between the share of seats it can optimally gerrymander to win and the seat share it would have received in a purely proportional election. A high gerrymandering power indicates there is a plan that stretches p's vote share

[9]As our intervals are symmetric about the middle point, this means Ds have a negative score.

into a disproportionately large number of districts. A low (or negative) gerrymandering power indicates p is unable to stretch its vote into many extra wins (or is unable to win a proportional amount of seats).

To gerrymander for party p while staying compact, we run our algorithm with the objective of generating plans which are as compact as possible while maintaining k wins[10] for party p. As we saw in previous sections, our algorithm is capable of generating highly compact districts and highly partisan districts. Unsurprisingly, we find it performs quite well when combining these goals. For example, in NC our algorithm can stretch the number of districts Rs win to all 13 using the 2016 election data, all while creating a plan more compact than the existing one (in the existing plan, Rs won 10 of 13). We can also create a map for Ds that is more compact than the existing one, but in which they win 11 seats instead of 3.

We vary k from the most partisan possible outcome (maximal number of districts won with no compactness constraint), to the most compact possible outcome (number of seats won by p in the Polsby-Popper compact plan from Section 6). PA's results can be seen in Figure 4. A full description of our method, and the figures for WI and NC, are in the Appendix.

7.1 Effect of Increasing Compactness

While increasing the required Polsby-Popper score lowers the gerrymandering power of both parties, to have an impact, a steep increase beyond what current the plans use (and often near the most compact) is required. Figure 4 shows the existing plans do not have a compactness score that constrains any party's gerrymandering (this is also true in WI and NC).

While compactness requirements can limit gerrymandering power, they are an unbalanced tool, favouring the R party. That is, for almost any Polsby-Popper score, their gerrymandering power is well above the D one. In PA there is no Polsby-Popper score where the Ds have an advantage. In NC and WI there is a brief period of near maximum compactness requirements where the Ds have the advantage, and even then it is small and quickly reverses. In WI when the compactness requirement is lower than that of the current plan the democrats do have a (very) marginal advantage, but more stringent requirements give a large R advantage. Measuring the average distance between the two curves, there is about 10% R advantage in gerrymandering power in PA and NC, and about 4% in WI.

Moreover, even with the most extreme compactness requirements, Rs are able to stretch their vote share beyond proportional, while Ds, even if they can have any legal districting they desire, have a negative gerrymandering power – they cannot even reach their proportional allocation.

8 Discussion

In this work we introduced a modular and powerful automated redistricting technique. We showed our technique can generate highly partisan districts, as biased as the hand drawn ones from 538. We also showed our method is able to generate compact districts, far more compact (according to various metrics) than the plans used in practice (or electoral experts' compact plans). Using our technique we showed how designing plans for compactness can reduce partisan bias over the existing plans. Despite their design objective, we found these plans still had a partisan bias. We then used our technique to design plans which are both partisan and satisfy compactness restrictions (according to the Polsby-Popper metric). We found while compactness restrictions can reduce the ability of either party to gerrymander, the potential for some degree of gerrymandering remains, and is, once again, with a partisan bias. All the biases we saw, in all

[10]By win we mean party p has a majority of votes.

states, under all metrics, with their differing characteristics, pointed towards an advantage to the geographic spread of Republican voters, i.e., a rural party advantage.

We see many applications of our algorithm for future work, but a few clear issues stand out. An immediate direction is to expand the gerrymandering power metric used in Section 7. In this work, we used the same one-vote-margin definition of victory used in previous work (whereby the party with more votes according to the last election is projected to win a given district). This can be extended to use at least a threshold win margin or to use probabilistic models (in which one can also incorporate the effect of a fraction of eligible voters abstaining from voting). We also believe the Republicans redistricting advantage deserves further exploration. This rural party advantage was noticed in simulations [Borodin *et al.*, 2019] and analytically argued by Rodden [2019]. To explore this relation we are working on modifications of the uniform swing model that incorporate ideas like population density. With these models we hope to generate vote distributions that resemble what is seen with the increasing urban-rural divide. Finally, since our algorithm is highly modular we believe it can be configured to design districts that are fair from a partisan perspective. While there are many definitions of "fair", our algorithm is capable of optimizing for any that can be expressed numerically and calculated from an arbitrary plan. An interesting future direction could be exploring the tradeoff in fairness and various other desirable goals, such as compactness.

References

Yoram Bachrach, Omer Lev, Yoad Lewenberg, and Yair Zick. Misrepresentation in district voting. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 81–87, New York, July 2016.

Amariah Becker and Justin Solomon. Redistricting algorithms. *Political Geometry*, 2021.

Bill Bishop. *The Big Sort: Why the Clustering of Like-Minded America is Tearing Us Apart.* Mariner Books, 2009.

Allan Borodin, Omer Lev, Nisarg Shah, and Tyrone Strangway. Big city vs. the great outdoors: Voter distribution and how it affects gerrymandering. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 98–104, Stockholm, Sweden, July 2018.

Allan Borodin, Omer Lev, Nisarg Shah, and Tyrone Strangway. Primarily about primaries. In *Proceedings of the 33rd Conference on Artificial Intelligence (AAAI)*, pages 1804–1811, Honolulu, USA, January-February 2019.

David Butler. The electoral process the redrawing of parliamentary boundaries in Britain. *British Elections and Parties Yearbook*, 2(1), 1992.

Amittai Cohen-Zemach, Yoad Lewenberg, and Jeffrey S. Rosenschein:. Gerrymandering over graphs. In *Proceedings of AAMAS*, pages 274–282, Stockholm, Sweden, July 2018.

Nate Cohn, Matthew Bloch, and Kevin Quealy. The new Pennsylvania congressional map, district by district. New York Times, February 19 2018.

Common Cause v. Lewis. 834 S.E.2d 425. Wake County Superior Court, November 2019.

Charlie Cook. The Cook political report. https://cookpolitical.com/, 2018. Accessed: 2021-01-01.

M.E. Dyer and A.M. Frieze. On the complexity of partitioning graphs into connected subgraphs. *Discrete Applied Mathematics*, 10:139–153, 1985.

E. J. Engstrom. Stacking the states, stacking the house: The partisan consequences of congressional redistricting in 19th century. *APSR*, 100:419–427, 2006.

Robert S. Erikson. Malapportionment, gerrymandering, and party fortunes in congressional elections. *The American Political Science Review*, 66(4):1234–1245, 1972.

Robert S Erikson. Malapportionment, gerrymandering, and party fortunes in congressional elections. *The American Political Science Review*, pages 1234–1245, 1972.

Dan S. Felsenthal and Nicholas R. Miller. What to do about election inversions under proportional representation? *Representation*, 51(2):173–186, 2015.

Benjamin Fifield, Michael Higgins, Kosuke Imai, and Alexander Tarr. A new automated redistricting simulator using markov chain monte carlo, January 2018.

John N. Friedman and Richard T. Holden. The rising incumbent reelection rate: What's gerrymandering got to do with it? *Journal of Politics*, 71:593–611, 2009.

Bernard Grofman and Gary King. The future of partisan symmetry as a judicial test for partisan gerrymandering after LULAC v. Perry. *Election Law Journal*, 6(1):2–35, 2007.

B. Grofman, W. Koetzle, and T. Brunell. An integrated perspective on the three potential sources of partisan bias: Malapportionment, turnout differences, & the geographic distribution of party vote shares. *Electoral Studies*, 16(4):457–470, 1997.

Bernard Grofman. Measures of bias and proportionality in seats-votes relationships. *Political Methodology*, pages 295–327, 1983.

Samuel Issacharoff. Gerrymandering & political cartels. *Harvard Law Review*, 116(2):593–648, 2002.

Omer Lev and Yoad Lewenberg. "reverse gerrymandering": a decentralized model for multi-group decision making. In *Proceedings of the 33rd Conference on Artificial Intelligence (AAAI)*, Honolulu, Hawaii, January 2019.

Justin Levitt. Where are the lines drawn? https://redistricting.lls.edu/redistricting-101/where-are-the-lines-drawn/, 2020.

Yoad Lewenberg, Omer Lev, and Jeffrey S. Rosenschein. Divide and conquer: Using geographic manipulation to win district-based elections. In *Proceedings of the 16th International Coference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 624–632, São-Paulo, Brazil, 2017.

League of Women Voters of Pennsylvania v. Commonwealth of Pennsylvania. Number 159 in MM 2017. Supreme Court of Pennsylvania, January 2018.

Alec MacGillis. Go midwest, young hipster. New York Times, October 23 2016.

MGGG. Metric Geometry and Gerrymandering Group; https://mggg.org, 2020. Accessed: 2021-01-01.

Rick Noack. The urban-rural divide that bolstered trump isn't just an american thing; it's prevalent in europe, too. *Washington Post*, November 27 2016.

Wesley Pegden, Ariel D. Procaccia, and Dingli Yu. A partisan districting protocol with provably nonpartisan outcomes. ArXiv:1710.08781, October 2017.

Clemens Puppe and Attila Tasnádi. A computational approach to unbiased districting. *Mathematical and Computer Modelling*, 48(9-10):1455–1460, 2008.

Jonathan Rodden. *Why Cities Lose: The Deep Roots of the Urban-Rural Political Divide*. Basic Books, 2019.

Rucho v. Common Cause. Number 18-422 in 588 U.S. Supreme Court of the United States, 2019.

P.H. Schuck. The thickest thicket: Partisan gerrymandering and judicial regulation of politics. *Columbia Law Review*, 87(7):1325–1384, 1987.

Erel Segal-Halevi, Shmuel Nitzan, Avinatan Hassidim, and Yonatan Aumann. Envy-free division of land. *Mathematics of Operations Research*, 45(3):797–1192, August 2020.

Nate Silver. 538: The gerrymandering project. https://fivethirtyeight.com/tag/the-gerrymandering-project/, 2020.

Andranik Tangian. Computational application of the mathematical theory of democracy to Arrow's impossibility theorem (how dictatorial are arrow's dictators?). *Social Choice and Welfare*, 35(1):129–161, June 2010.

René van Bevern, Robert Bredereck, Jiehua Chen, Vincent Froese, Rolf Niedermeier, and Gerhard J. Woeginger. Network-based vertex dissolution. *SIAM Journal on Discrete Mathematics*, 29(2):888–914, 2015.

William Vickrey. On the prevention of gerrymandering. *Political Science Quarterly*, 76(1):105–110, 1961.

Samuel S.-H. Wang. Three tests for practical evaluation of partisan gerrymandering. *Stanford Law Review*, 68:1263–1321, June 2016.

David Wasserman. Hating gerrymandering is easy. fixing it is harder. FiveThirtyEight, 25 January 2018.

Archer Wheeler and Philip N. Klein. The impact of highly compact algorithmic redistricting on the rural-versus-urban balance. In *SIGSPATIAL '20: 28th International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, November 3-6, 2020*, pages 397–400, 2020.

A Appendix

A.1 Description of the GREAT Algorithm

In this section we will describe the GREAT algorithm in its general form.

Our basic approach is built around simulated annealing (SA), a general optimization technique that has found much success in various discrete optimization problems. To build our SA approach we will expand on the Markov Chain Monte Carlo (MCMC) package for redistricting known as *Gerrychain* provided by MGGG. Details of what they provide and our extensions will be expanded on in subsequent sections.

With hill-climbing based approaches, at every iteration we have a candidate, current, solution and we examine some neighbouring solution. We need to decide if we should move to this neighbouring solution or not. In standard hill-climbing methods, like (greedy) local search, only moves which improve the solution are accepted.

At a high level, simulated annealing-based optimization is essentially a hill-climb, but moves are allowed towards inferior solutions. The ability to accept non-improvement steps, i.e., objectively worse solutions, becomes increasingly less permissible as the optimization proceeds. The logic with non-improvement steps is that they allow the procedure to escape local optima earlier in the process. If the space of solutions is non-convex (with respect to solution quality) these local optima can act as sinks for procedures which only allow improvement steps. The ability to accept a non-improvement neighbour is controlled by two parameters, the temperature of the system and the difference in quality (also known as energy-difference) of the current and proposed solution.

Energy: The first component of a SA based approach is the *energy* of a solution. The energy of a solution is a function which maps a potential solution to a numeric measure of quality, for our work we consider the set of solutions to be all legal districting plans. That is if a graph of a state has node set with n nodes and they must be partitioned into K districts the energy function is:

$$E : [n]^K \to \mathbb{R}_+ \cup \infty. \tag{1}$$

It is standard for lower energy values to correspond to superior solutions and for zero energy to be the best any solution can take on[11].

Proposal: The second component of the SA based approach is the *proposal* function. A proposal function P takes in a potential solution S and picks a *neighbour* S' of S:

$$P : [n]^K \to [n]^K. \tag{2}$$

There is no fixed definition of what a neighbour is and this can vary from domain to domain, or even within a problem itself. For our work we will use the following recom-proposal function, which was first suggested by MGGG. The recom-proposal is presented in the following algorithm 1:

When we say a solution or district is valid we mean that it satisfies all constraints we place on districts. Note that using Kruskal's algorithm for drawing spanning trees (the drawing of T_t) can effectively be done in time linear in the number of nodes left in R[12].

[11] The optimal solution for a particular instance could have non-zero energy, zero just serves as a lower bound. Generally invalid solutions have infinite energy.

[12] To draw a random spanning tree we find the minimum spanning tree after randomly assigning each edge a weight.

Algorithm 1 recom_proposal(G, S, j):

1: Let S be the current solution (districting plan which partitions the vertices into k components).
2: Pick $i \in \{2, \cdots, j\}$ random, but connected, districts from S (where $j \leq k$).
3: Let R denote the precinct nodes in these i districts.
4: **for** $t \in \{1, \cdots, i-1\}$ **do**
5:     Draw a random spanning tree using only the nodes of R. Call this spanning tree T_t.
6:     Sample a random edge e that has yet to be picked (see particular heuristic in text) in T_t. This divides T_t into 2 connected components.
7:     If T_t beneath e forms a valid district: Make it one of our new districts, remove these nodes from R.
8:     Else, if there are edges yet to be sampled and T_t beneath e is not a valid district: Repeat step 6.
9:     If all of the edges of T_t have been sampled and no valid district was ever found in T_t: repeat step 5.
10: **end for**
11: Let the remaining nodes of R be the final new district.
12: Let S' be the solution identical to S but where R has been redistricted according to steps $4 - 11$.
13: **if** S' is a valid solution **then**
14:     Return S'
15: **else**
16:     Retry the algorithm from step 1.
17: **end if**

Each time through the for loop at step 6 the algorithm may pick several edges in the spanning tree T_t (steps 6 through 8). For each iteration of the loop the first such e must be connected to a leaf of T_t (the first time step 6 is executed for each loop iteration). If the node under e is not a valid district then another random edge is chosen (step 8). This next edge is either another edge connected to a leaf, or the edge directly above e in T_t. In subsequent steps (if they are required) the algorithm picks an edge e that has not been previously selected in T_t. In addition there are two more restrictions on selecting the new e: First, the algorithm requires that this edge is either connected to a leaf node or is the direct ancestor of a previously chosen edge. Second, the algorithm requires that all of the edges under e in T_t have previously been selected. Intuitively, this process works by bubbling up through the various branches for T_t, trying edges until a sufficient one is found.

Each time through the for loop the algorithm should find one of the districts we need. It is possible some iteration of the for loop will fail to find a valid division, sampling every edge in T_t (step 9). In this case this iteration of the for loop restarts, finding a new spanning tree, but keeps the districts found up to this point.

It is also possible that at some iteration of the for loop, no spanning tree can lead to a valid districting. That is, it will just draw new spanning trees forever. We are unaware of any method that can detect this scenario, short of sampling every spanning tree. As a heuristic solution, we put a time limit on the algorithm. We found the algorithm tends to find solutions within 20 seconds for the most complex instances we work with. If after 1000 seconds we do not have a solution we restart the entire algorithm. This was an addition we made to the functionality provided by MGGG.

We note that the recombination method proposed by MGGG only worked for recombining two districts at a time, whereas we extended it to work for any number. In the step where we pick i random districts for recombination we do so by sampling uniformly at random from the set of all sets of connected districts up to size j. The intention of the MGGG method seems to be the same (for $j = 2$), but their code shows that they pick districts by uniformly sampling from all edges which cross district boundaries. This will favour picking pairs of districts which share large boundaries (in terms of nodes). In general we found that increasing the number of merged districts beyond two did not improve our solution quality (but it did slow the procedure down).

Temperature: The third part of the SA approach is the *temperature*, which acts as a control for how likely negative moves are at a given state of time. Generally the temperature is a decreasing function of the number of iterations so far in the optimization. While there are many temperature functions and choosing the ideal one is somewhat of a black-box in optimization, we've found the following temperature function works well (here (s) is an iteration counter):

$$T(s) = 10000 \cdot (0.99^s) \tag{3}$$

This is known as the exponential cooling schedule. From the initial temperature of $10,000$ at every step we retain 99-percent of the remaining heat until we eventually cool to a temperature of 0.

A.1.1 The simulated annealing method

The simulated annealing method is as follows for a graph $G = (V, E)$ which is to be partitioned into K districts:

Algorithm 2 simulated_annealing_for_gerrymandering(G):

Let $S_0 = recom\_proposal(G, None, K)$.
$i = 0$
while $i \leq s_{max}$ **do**
    $S' = recom\_proposal(G, S_i, j)$
    if $E(S_i) \geq E(S')$ **then**
        $S_{i+1} = S'$
        $i = i + 1$
    else
        Let $\Delta E = E(S_i) - E(S')$
        Let r be drawn uniformly at random from $[0, 1]$.
        if $\exp \frac{\Delta E}{T(i)} \geq r$ **then**
            $S_{i+1} = S'$
            $i = i + 1$
        end if
    end if
end while

In the first step $None$ refers to the districting which makes no assignments. To find the initial partition we do not need to provide the sub-routine with a valid districting since we are recombining all of the nodes. Intuitively the algorithm will always move to a lower energy solution and will move to a higher energy solution with high probability if the increase in energy is not too high and the temperature is not too cool.

It is entirely possible that the procedure will eventually be caught in a local optimum (or even the global one) it cannot move away from with reasonable probability. This is especially true later on as the temperature cools. If this is the case the main loop will, with very high probability, make no progress to completion. Because of this we often set a hard time limit and cut off the procedure after this point. In general with SA, or any random algorithm, one needs to run many parallel executions of the procedure, and each of these will iterate over many potential solutions. The best of all iterated solutions will be chosen as the returned solution.

A.2 Description of 538 Reconstruction

Reconstructing the model 538 used to evaluate wins involved multiple steps and analysis.

A.2.1 The Cook PVI

538 builds their probabilistic model on the Cook partisan voting index (Cook PVI or PVI) published by the Cook Political Report [Cook, 2018] a non-partisan and independent newsletter that analyzes elections and trends in the United States. The PVI is a metric which measures how partisan a group of voters, in particular those who form a congressional district, are relative to the average voter in the United States. To calculate the PVI there needs to be a running value for how partisan the country is as a whole (call this value β_D). To calculate this we take the number of votes garnered in the two most recent presidential elections[13] and see what fraction of these votes belong to the Democratic party. The partisan skew expresses the average of the vote fractions for the Democrats in the last two presidential elections. Note, this is an average of averages, it is not weighted by the total votes in each election. To calculate the PVI 538 used we need the 2012 election where:

- For the Barack Obama and Joe Biden of the Democratic party $65,915,795$ votes.
- For Mitt Romney and Paul Ryan of the Republican party $60,933,504$ votes.

For the 2016 election the exact results were:

- For the Hillary Clinton and Tim Kaine of the Democratic party $65,853,514$ votes.
- For Donald Trump and Mike Pence of the Republican party $62,984,828$ votes.

Using the above information we get the value of β_D would be:

$$\frac{\frac{65{,}915{,}795}{65{,}915{,}795+60{,}933{,}504} + \frac{65{,}853{,}514}{65{,}853{,}514+62{,}984{,}828}}{2} \tag{4}$$

Thus, we see β_D is roughly 51.53857559136132%. Note because the United States uses the electoral college system, Donald Trump and Mike Pence won the 2016 election despite receiving fewer votes than Hillary Clinton and Tim Kaine. Note that while the United States is effectively a two party system there are other candidates who run for various offices including president. For example, in 2016, Gary Johnson and Joe Weld of the Libertarian party received $4,489,341$ votes (over 3% of the total vote). Since the Cook PVI is meant to be a direct comparison between the Democratic and Republican party it does not factor in third-party votes. [14]

The PVI of a district is then just how partisan that district is relative to β_D. In district i, let the total number of Democratic votes denoted by $N_i^{D,1}$ and the Republican ones as $N_i^{R,1}$ for the last presidential election; and $N_i^{D,2}$ and $N_i^{R,2}$the same for the presidential election before that, then the PVI is :

[13] The presidential election is chosen since they use the same candidate for the entire country and thus are free of any local effects.

[14] The Cook Political Report does not actually publish the formula or exact method for this metric. In particular they did not make it clear if the mean of the two elections was weighted or not. We confirmed our interpretation by measuring the reported PVI in single district states and comparing to the formula we derived.

$$100 \cdot \left(\frac{\frac{N_i^{D,1}}{N_i^{D,1}+N_i^{R,1}} + \frac{N_i^{D,2}}{N_i^{D,2}+N_i^{R,2}}}{2} - \beta_D\right) \tag{5}$$

Equation (5) can range from $-\beta_D \cdot 100$ for completely Republican dominated districts, to $(1-\beta_D) \cdot 100$ for districts with only Democratic voters, or 0 for districts which match the national average in the last two presidential elections.

Intuitively a district with a very positive PVI should be safely Democratic. Even if there is a uniform swing towards Republican sentiments this particular district should lean Democratic (the same is true for Republicans and districts with a very negative PVI).

A.2.2 The 538 Sigmoid

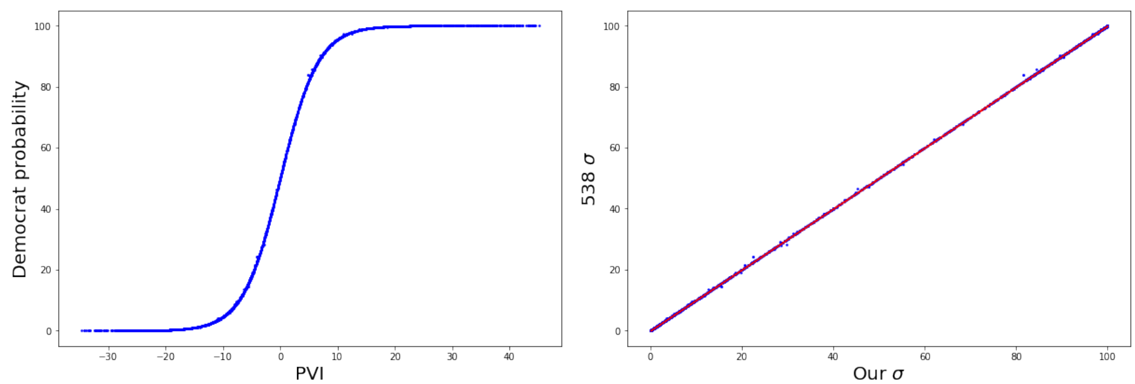

Figure 5: First figure shows the reported Cook PVI for each district created by 538 vs their estimation of the probability that the Democrats will win that district. Second figure shows the output of our reconstruction of the 538 model vs the 538 model itself, the inputs to these two models were each of the districts created by 538.

The next step in the 538 model is going from the Cook PVI for a hypothetical district to how probable it is that district elects a Democrat. We suspect 538 went with the sigmoid function (the sigmoid function is ideal since it is monotone in its inputs and the output falls in the range $(0,1)$). Recall, the sigmoid function takes the form:

$$\sigma(x) = \frac{1}{1+e^{-w \cdot x}} \tag{6}$$

This function is fitted to data (x) by adjusting the weight parameter w. Unfortunately 538 was not specific on what exact data was used to fit the sigmoid, or if regularization terms were included in the fitting. Luckily, 538 did publicly report the Cook PVI and their derived probability of a Democratic win for all the districts in their catalogue for each state[15]. The probability of a Democratic win, plotted against the Cook PVI (first subfigure of Figure 5), clearly shows a sigmoid shape. From here we just need to derive what weight parameter w they use is. To figure this out we first invert the sigmoid function using the log-odds (or logit) function:

$$logit(\sigma(x)) = \log_e \frac{1}{1-\sigma(x)} \tag{7}$$

[15] In total there are 2568 districts. These districts are the entirety of all of their created plans. This includes plans such as the partisan plans, competitive plans and plans that emphasize compactness. They also include the current congressional plans.

Inverting the sigmoid function with the logit function would produce a line given by $y = w \cdot x$, thus we simply need to invert any two data point in the second subfigure of Figure 5 and take the slope of the resulting line as w (since this is a linear function of one variable any two distinct points are sufficient to determine it). Briefly, we mention two important points. Firstly, the sigmoid, and hence the line from the logit, may have a bias term associated with them. We found 538 did not include one since their sigmoid passes through $(0, 50)$[16] and the resulting logit line passes through $(0, 0)$. Secondly, the points 538 published do not perfectly follow a sigmoid, instead there is a small amount of "jitter" on some of the points in the first subfigure of Figure 5. This deviation could simply be a rounding issue or minor transcription errors, in either case the points still very closely follow the sigmoid pattern. Because of the small amount of noise the resulting inverted plot found with the logit function will not be perfectly linear. Thus our choice of the two points for the inference of w would (very slightly) influence the outcome. To mitigate this issue we take the ordinary least squares (OLS) regression line, also known as the line of best fit, for all of the points (after inverting them with the logit). We found the slope of the OLS line was 0.3047121945377743 which we ended up using for the w parameter in our sigmoid model. Our resulting model is a near perfect fit for the 538 model since they form the line $y = x$ when plotted against each other (the final subfigure of Figure 5)

A.3 Additional Material for Section 5

In this section we describe how to use Algorithm 2 to match the 538 partisan plans. The main specifics required are the specifications of Equation 1. For our purposes we will have our energy function be based on the expected number of districts won with one slight modification. Say we are gerrymandering for the Democratic party. If a potential solution S is comprised of K districts called $S_1, \cdots, S_K$ then the energy of that solution is:

$$E(S) = K - \sum_i v_D(S_i), \tag{8}$$

where $v_D(S_i)$ is equal to :

$$v_D(S_i) = \begin{cases} \sigma(S_i) & \sigma(S_i) \leq \tau \\ 1 & otherwise \end{cases}$$

Where σ is the sigmoid function we derived from the 538 data. If the target party is Republican party we can replace $v_D(S_i)$ with $v_R(S_i)$ [17] which is defined as follows:

$$v_R(S_i) = \begin{cases} 1 - \sigma(S_i) & 1 - \sigma(S_i) \leq \tau \\ 1 & otherwise \end{cases}$$

Here, τ is the threshold for what we consider a strong win. That is if a district win probability for our target party is above τ, we say that is a safe win for that party. Intuitively our function is aiming to maximize the number of safe wins for the target party. Our method would prefer a solution with several borderline safe wins over a solution with fewer very safe wins. For all of our simulations we copy 538 and use $\tau = 0.82$.

Our only constraints were that districts must be connected and within half a percent of the ideal population. For each state and party we found well before our cutoff of 24 hours the algorithm had stopped making steps. We also found with 60 parallel runs we were able to get the results shown in Table 2.

[16] There is exactly one data point with a PVI of 0 and a Democratic probability of winning of 50%.

[17] Recall, the probability the Republican party wins a district is just one minus the probability the Democratic party wins it.

A.4 Additional Material for Section 6

Compactness presents a set of challenges, from definitions to implementation.

A.4.1 The Compactness Measures

Here we formally define each of the compactness metrics we use. For ease of notation we define the following functions for a district i with subgraph G_i and polygon P_i. Let the area be $A(P_i)$, let the length of the perimeter be $L(P_i)$, and let the geometric centre point be $C(P_i)$[18]. These measures can be defined for any arbitrary $2D$ polygon (not necessarily just those that belong to a district). Let the straight line distance between two points $a, b \in \mathbb{R}^2$ be $d(a, b)$. And finally, let the shortest (in terms of number of edges) path between two nodes in $i, j \in V(G_i)$ be $SP(i, j)$.

Polsby-Popper (PP) Let C_i be the circle where $L(C_i) = L(P_i)$. The Polsby-Popper score is equal to $\frac{A(P_i)}{A(C_i)}$. This value ranges from the least compact 0 (a district with no area), to the most compact 1 (a circle-shaped district).

Convex Hull (CH) Let CH_i be the convex shape which bounds P_i and has the minimal value for $A(CH_i)$. The Convex Hull score is equal to $\frac{A(P_i)}{A(CH_i)}$. This value ranges from the least compact 0 (a district with no area), to the most compact 1 (a convex district).

538 metric The 538 metric was not formally explained, it is described as "the average distance between each constituent and his or her district's geographic centre". One possible interpretation of this could be $\sum_{\forall u \in V(G_i)} d(u, C(P_i))$. But it is also possible there are other interpretations of what centre means.

A.4.2 Redistricting for Compactness

First we describe how Equation 1 is set to optimize for each of these metrics. For the Convex Hull and Polsby-Popper metrics, given a plan we calculate the compact score in each district (in the range $[0, 1]$) and take the overall mean of them, call this value x. We then set the energy of this plan to be $1 - x$. Since 538 calculated the solution for the final metric, we do not optimize for it. From what we've found[19], the 538 optimization also seemed to be annealing based, but instead of a tree based recombination the proposals involved swapping nodes on the boundaries of districts.

For each of our measures we ran 120 threads in parallel (this led to many threads ending up with nearly the same highest level of compactness) for 48 hours. We found this time more than sufficient for ensuring the process stopped making improvement steps.

A.4.3 Additional Plots for Swing Advantage

Here we provide the plots for other data ranges and elections showing the Republican advantage in the uniform swing model. In Section 6, Figure 3 showed the advantage for the 2012 election using the $[40, 60]$ data range. Here we provide the other three items mentioned. The 2016 election for both the $[40, 60]$ data range (Figure 6) and $[45, 55]$ data range (Figure 7). And the 2012 election using the $[45, 55]$ data range (Figure 8). As mentioned earlier, in any setting the partisan bias of the compact plans is lower than that of the 2011 plans, with one exception. In Wisconsin in 2016, over the $[45, 55]$ data range the Convex Hull

[18] This is the point where P_i would balance on a pin tip.

[19] See https://bdistricting.com/2010/ for a high level description of their method.

plan has a marginally higher bias than the 2011 plan. Furthermore every single plan we examine shows a Republican bias, none have a democrat lean.

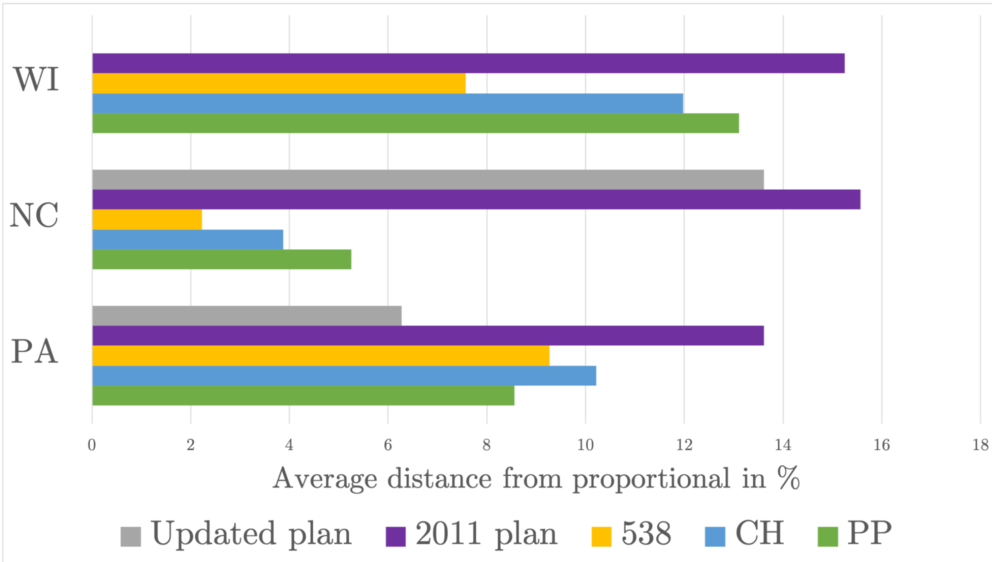

Figure 6: Average distance from the R swing curve to the $y = x$ line over the range $[40, 60]$ for the various plans in each state using 2016 presidential election data. The WI 2011 plan was not struck down, unlike in PA and NC, thus there is no "new" plan for it.

A.5 Additional Materials for Section 7

In Section 7 we discussed combining gerrymandering with compactness constraints, which we further elaborate on below.

A.5.1 Compact Gerrymandering

We use our method to generate compact, but partisan, districts. First, for each party, we generate highly partisan outcomes. That is, given a partitioning of the nodes of G into $S = (S_1, \cdots, S_K)$ set Equation 1 as follows (assuming we are gerrymandering for the Democrats):

$$E(S) = K - \sum_i v_D(S_i), \tag{9}$$

where $v_D(S_i)$ is equal to :

$$v_D(S_i) = \begin{cases} \frac{N_i^D}{N_i^D + N_i^R} & \frac{N_i^D}{N_i^D + N_i^R} \leq \tau \\ 1 & otherwise \end{cases}$$

Here N_i^D is the total Democrat vote in district i (N_i^R is the total Republican vote in district i). If we want to gerrymander for the Republicans, replace $v_D(S_i)$ with $v_R(S_i)$ which is defined as follows:

$$v_R(S_i) = \begin{cases} \frac{N_i^R}{N_i^D + N_i^R} & \frac{N_i^R}{N_i^D + N_i^R} \leq \tau \\ 1 & otherwise \end{cases}$$

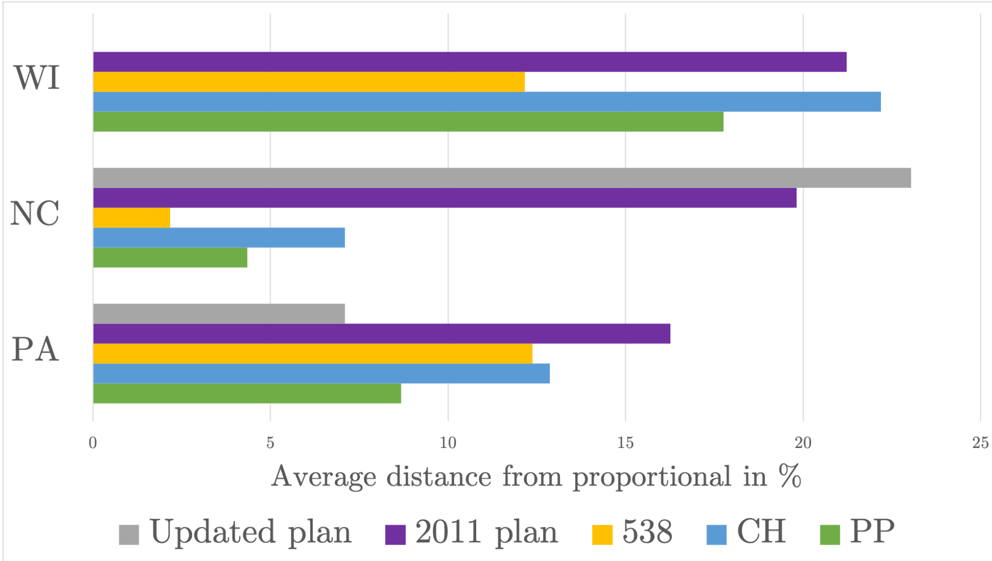

Figure 7: Average distance from the R swing curve to the $y = x$ line over the range $[45, 55]$ for the various plans in each state using 2016 presidential election data. The WI 2011 plan was not struck down, unlike in PA and NC, thus there is no "new" plan for it.

For our experiments we always take $\tau = 0.5$, that is we require a simple majority of the vote for a win. This is similar to our method for emulating 538, but now the sigmoid function's contribution to the energy has been replaced by a linear distance to winning the district. We tried other definitions of $v_D(S_i)$ and $v_R(S_i)$, such as exponentially decreasing energy as one gets closer to winning the district, a small decreasing contribution to energy even if the target party is winning a district, and modifications of the sigmoid. In the end we found the presented definitions worked best.

For the rest of this section, assume we are gerrymandering for party P. For our first phase, in each state for P we run our method 288 times for 48 hours. This time limit was more than sufficient for the convergence of the various processes. This first phase gives us several runs that have the most possible wins for P in each state, call this set of solutions W_{max}. In addition, for each state we ran 288 executions of our code to optimize for the Polsby-Popper score (just like we did in Section 6[20]) Then for P in each state, we have a range of potential win values $\{w_{max}, \cdots, w_{min}\}$ (where w_{max} is the most number of wins found for P in the first phase and w_{min} is the number of wins for P in the optimally compact solution).

For each value $w \in \{w_{max}, \cdots, w_{min} + 1\}$ we execute the following procedure. Dividing the solutions of W_{max} among 288 cores as evenly as possible we run our algorithm using these solutions as the initial plans and optimize for the Polsby-Popper score (again like in Section 6). We add the additional constraint, for a proposal to be considered in algorithm 1 we require that the total number of wins for P is at least w. We found improvements were ending well before the 24 hour cutoff for each of these simulations.

A.5.2 Missing Charts for Gerrymandering Power

In Section 7 we explored what happened to the gerrymandering power as stronger Polsby-Popper constraints were added. We found compactness constraints can decrease gerrymandering power, but to have any impact

[20]For these new 288 runs we included them with the results reported in that section. They did not lead to a significant improvement.

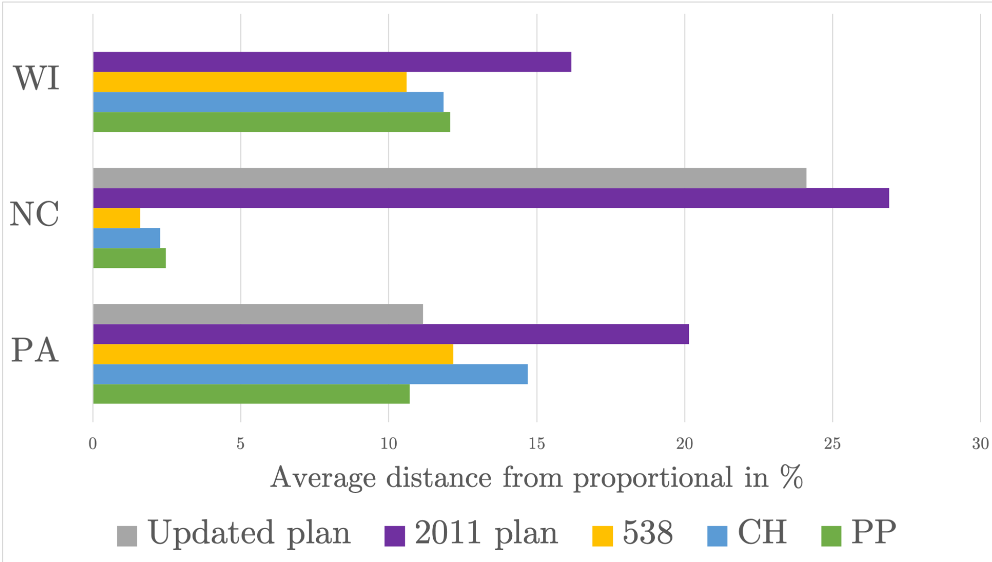

Figure 8: Average distance from the R swing curve to the $y = x$ line over the range $[45, 55]$ for the various plans in each state using 2012 presidential election data. The WI 2011 plan was not struck down, unlike in PA and NC, thus there is no "new" plan for it.

on any party in any state, the required levels were beyond the compactness of current plans. We provided the plot showing this effect in PA for the 2012 election, here we provide the same plots for WI and PA (which show a similar effect). Figure 9 shows the equivalent plot for WI in 2016. Note, since the 2011 plan was never struck down there is only one vertical line. Figure 10 shows the equivalent plot for NC in 2016. As was the case with PA, the 2011 plan in NC was found to be illegal. It was replaced in 2016.

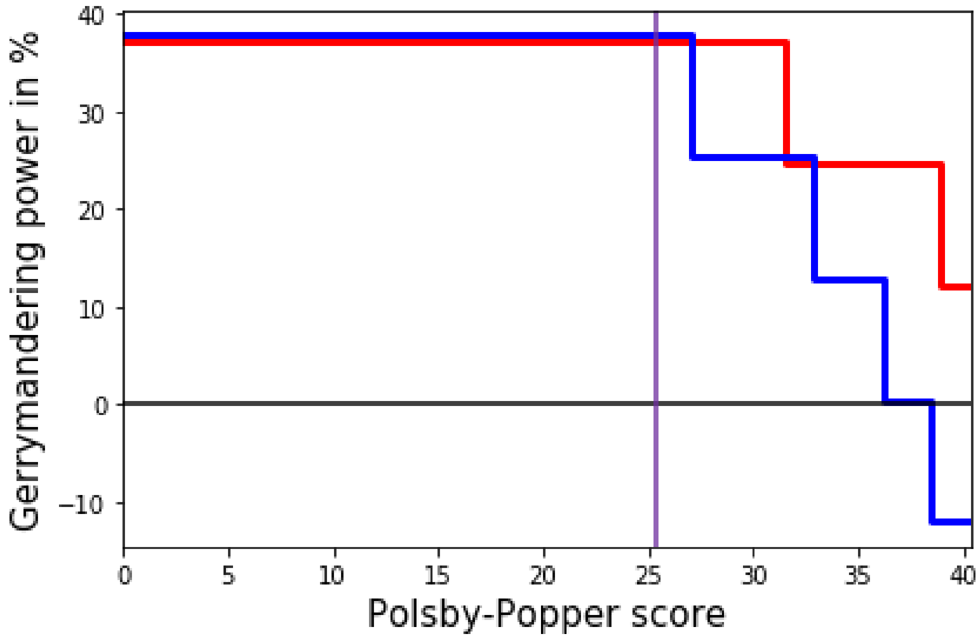

Figure 9: Gerrymandering power when faced with a minimum required Polsby-Popper score using data from the 2016 WI presidential election. R in red; D in blue. The vertical purple line is the Polsby-Popper score of the 2011 congressional plan. Average distance between the two curves is a 4.3% advantage for the Rs.

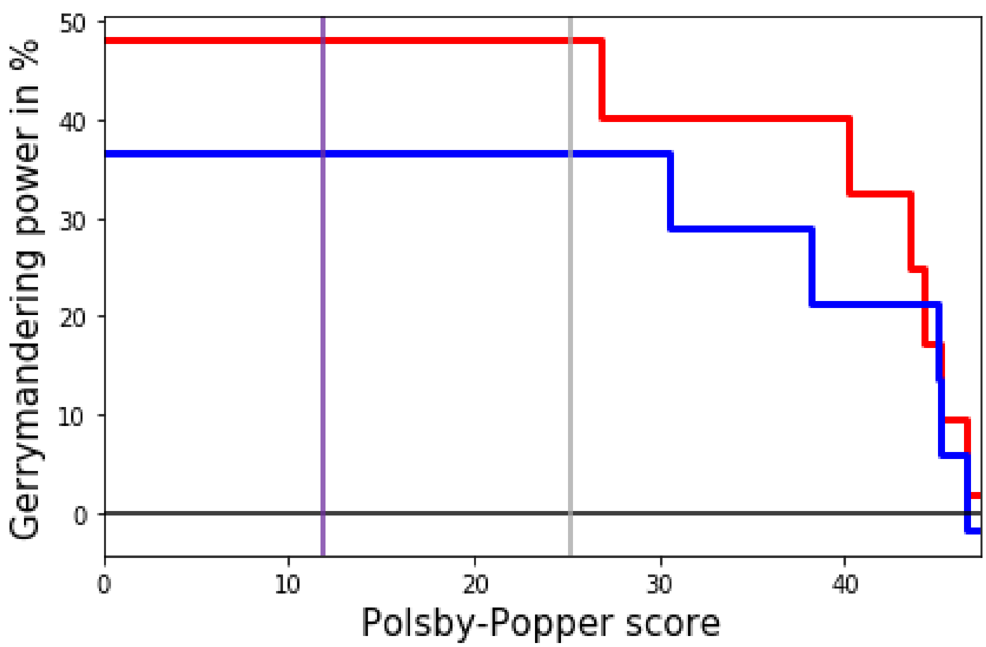

Figure 10: Gerrymandering power when faced with a minimum required Polsby-Popper score using data from the 2016 NC presidential election. R in red; D in blue. The vertical purple line is the Polsby-Popper score of the 2011 congressional plan, the vertical grey line is the Polsby-Popper score of the 2016 court mandated plan. Average distance between the two curves is a 10.4% advantage for the Rs.