Combinatorial Optimization Problems Arising from Graph-Based Models

by

Samuel Boardman

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Applied and Interdisciplinary Mathematics)
in The University of Michigan
2026

Doctoral Committee:

Professor Mahdi Cheraghchi, Co-Chair
Professor Martin Strauss, Co-Chair
Professor Euiwoong Lee
Professor David Speyer

Samuel Boardman
stboard@umich.edu
ORCID iD: 0009-0009-1958-2534

© Samuel Boardman 2026

ACKNOWLEDGEMENTS

I would like to thank my two advisors, Martin Strauss and Mahdi Cheraghchi, for their consistent support in my development as a student and a scholar. Martin really encouraged me to explore my ideas at a time when I was unsure of my place within applied mathematics. I valued his openness to using real-world intuition—while still scrutinizing it—and the many enriching conversations that resulted. Mahdi, before serving as my advisor, taught me via two excellent courses that solidified my interest in theoretical computer science, with lots of banter and a sense of community. These positives translated seamlessly to the advising setting, and Mahdi's willingness to have wide-ranging discussions truly improved my experience.

I am also grateful for the remaining members of my committee, Euiwoong Lee and David Speyer. Euiwoong was always willing to chat and generated formidable ideas during our sessions. David was very accommodating with me and helped me identify improvements in the presentation of my work.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS ii

LIST OF FIGURES v

LIST OF TABLES viii

LIST OF ACRONYMS ix

ABSTRACT x

CHAPTER

1 Introduction 1

2 Influence Maximization with Signed Information 5

2.1 Introduction 5
2.2 The Model 7
2.2.1 Intuitive Description 7
2.2.2 Converting Examples into Beliefs 8
2.2.3 Formal Statement 10
2.3 Results 12
2.3.1 The Deterministic Case of Info Bal 12
2.3.2 The Stochastic Case of Info Bal 19
2.3.3 Hardness of Const Info Bal and the Stochastic Case of Info Bal 23
2.3.4 Simulation Outcomes 32

3 Ranking with Partitioning 35

3.1 Introduction 35
3.2 The General Case of the Problem 37
3.3 Algorithms for Special Cases 45
3.3.1 Complete Graph Case–Bounding Rank and Percentile 45
3.3.2 Linear Component Case 51
3.3.3 Uniform Value Case 54
3.4 Variants of the Problem 56
3.4.1 Equivalence Class Variant 56
3.4.2 Hierarchical Category Variant 58
3.4.3 Grid Variants 62

3.4.4 Applications of Recursive Algorithms for Rank and Percentile . . . 69
3.5 Numerical Experiments . . . 70

4 Redistricting with Hierarchical Rectangles . . . 76

4.1 Introduction . . . 76
4.2 Contiguous Redistricting and Proportional Representation . . . 77
4.3 Redistricting with (Hierarchical) Rectangles . . . 79
4.3.1 Worst-Case Gaps with Proportional Representation . . . 82
4.3.2 Solving Gerrymandering with Hierarchical Rectangles . . . 87
4.3.3 The Redistricting Game . . . 89
4.4 An Urban-Rural Model . . . 90

5 Conclusion . . . 105

APPENDIX . . . 111

BIBLIOGRAPHY . . . 118

LIST OF FIGURES

FIGURE

2.1 An instance of Info Bal with 4 nodes and probability parameter $p_e = 1$ for each edge e. Each node v_i has bias value b_i written inside the corresponding circle. Observe that seeding: 4 packets of $+1$ to v_1; 11 packets of -1 to v_2; 2 packets of $+1$ to v_3; and 8 packets of -1 to v_4 induces a resultant bias vector of $b' = (0, 0, 0, 0)$, hence minimizing each objective function ($l_1(b') = l_2^2(b') = l_\mu(b') = 0$) and solving the corresponding version of Info Bal for this instance. Note that we seed v_1 and v_4 the exact net number of packets required to neutralize their bias since neither has an incoming edge. On the other hand, v_2 must be seeded additional packets since v_1 is neutralized in the opposite direction that v_2 is, whereas v_3 is seeded fewer packets since v_2 is neutralized in the same direction that v_2 is (and the change from v_1's packets is already accounted for in the packets seeded to v_2). 11

2.2 A template graph for a potential member of each family of graphs described in the proof of Theorem 2.3.1, with each node's initial bias value written inside the corresponding circle. For this to be a member of the first family (for $d = 2$), the rightmost vertex and edges incident to it are deleted (in which case $n = 10$); for the second family (for $n = 20$), the rightmost vertex is replaced with 10 vertices adjacent to the vertices in the last layer. To generate additional graphs in the first family satisfying the first part of Theorem 2.3.1, the size of each layer is incremented. To generate additional graphs in the second family satisfying the second part of Theorem 2.3.1, the number of layers is incremented. 14

2.3 The reduction from X3C to Const Info Bal in Theorem 2.3.10, depicted with $m = 5$ and $n = 6$ ($k = 2$). Each node on the left has bias 0, and each node on the right has bias -1. 24

2.4 The reduction from X3C to Info Bal in Theorem 2.3.13, depicted with $m = 5$ and $n = 6$ ($k = 2$). Each node on the left has bias -1, each node in the center has bias -1, and each node on the right has bias 0. 27

3.1 A map depicting each municipality in New England by black boundary lines [7, 37]. There are many ways one could have partitioned these municipalities into contiguous states, each way carrying implications for how people may think about the relative standing of its constituent parts under some metric. 39

3.2 A visual representation of Algorithm 1 for unpaired intermediate elements used by an optimal algorithm. Our algorithm completes the largest half of the intermediate elements with tiny elements and pairs the remaining half with each other (among the intermediate elements not paired by the optimal algorithm), guaranteeing they cross the $\mu(S^*)$ threshold without relying on further tiny elements. 50
3.3 A partition that minimizes both the rank and rank percentile of 2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs, as recorded in Table 3.1. Subsets included in the partition are shaded. 73
3.4 A partition that maximizes the rank of 2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs, as recorded in Table 3.1. Subsets included in the partition are shaded. 74
3.5 A partition that maximizes the rank percentile of 2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs, as recorded in Table 3.1. Subsets included in the partition are shaded. . 75

4.1 The construction of V for Theorem 4.3.7 with $N \geq 16$ (at least four 4×4 subgrids), where $T = 1$. Shaded circles indicate Player 1's loyalty fraction is greater than 0.5; circle area is proportional to population. For values of N larger than 16, the pattern extends, as denoted by the ellipses. 84
4.2 The construction of V for Theorem 4.3.8 with $N \geq 6$, where $T = 1$. Shaded circles indicate Player 1's loyalty fraction is greater than 0.5; circle area is proportional to population. For values of N larger than 6, the pattern extends, as denoted by the ellipses. 87
4.3 A solution to an instance of Gerrymandering with Hierarchical Rectangles, with the resulting districts denoted as R_i (districts won by Player 1 are written in bold). Precincts are colored according to the value of p_1 at them, with values greater than .5 blue, values less than .5 red, and values equal to .5 white). $V_- = \emptyset$. Player 1's slate is $N_1 = 5$. Parameters used: $L = 50, W = 50, N = 10, C = \frac{7}{8}W, \delta = \frac{24}{470W}, \rho = .01$. 102
4.4 A solution to an instance of Gerrymandering with Hierarchical Rectangles, with the resulting districts denoted as R_i (districts won by Player 1 are written in bold). Precincts are colored according to the value of p_1 at them, with values greater than .5 blue, values less than .5 red, and values equal to .5 white). $V_- = \emptyset$. Player 1's slate is $N_1 = 13$. Parameters used: $L = 50, W = 50, N = 25, C = \frac{7}{8}W, \delta = \frac{24}{470W}, \rho = .01$. 102
4.5 A solution to an instance of Gerrymandering with Hierarchical Rectangles, with the resulting districts denoted as R_i (districts won by Player 1 are written in bold). Precincts are colored according to the value of p_1 at them, with values greater than .5 blue, values less than .5 red, and values equal to .5 white). $V_- = \emptyset$. Player 1's slate is $N_1 = 27$. Parameters used: $L = 50, W = 50, N = 50, C = \frac{7}{8}W, \delta = \frac{24}{470W}, \rho = .01$. 103

4.6 A solution to an instance of GERRYMANDERING WITH HIERARCHICAL RECTANGLES, with the resulting districts denoted as R_i (districts won by Player 1 are written in bold). Precincts are colored according to the value of p_1 at them, with values greater than .5 blue, values less than .5 red, and values equal to .5 white. $V_- = \emptyset$. Player 1's slate is $N_1 = 14$. Parameters used: $L = 100, W = 100, N = 25, C = \frac{7}{8}W, \delta = \frac{24}{470W}, \rho = .01$. 104

LIST OF TABLES

TABLE

2.1 The mean (over 10 trials of CONST INFO BAL) of observed l_2^2 objective function values for the given dataset using Equation (2.13). Regression analysis of these values suggests quadratic decay with r^2 values of .9997 (Gnutella) and .9994 (Irvine), with regression coefficients all statistically significant at the .001 level. 33

2.2 Observed changes in the l_2^2 objective function upon applying the given heuristic approach. 34

3.1 Minimum and maximum rank and percentile (rounded to 2 decimal places) for select EPA Common Reporting Tables (CRT) categories over all valid hierarchical category partitions. 72

4.1 Parameters and calculated bounds for varying grid dimensions and district counts. 101

LIST OF ACRONYMS

CRT Common Reporting Table

PR Population Regularity

ABSTRACT

We propose and analyze several graph-defined combinatorial optimization problems inspired by multiple application areas.

First, we consider an influence maximization model that uses the independent cascade approach, but allows two types for packets of information, $+1$ and -1. These represent the change in a receiver's sentiments on a product or belief on an empirical question (regarded as their "bias"), which can be justified via Bayes' rule. The algorithmic problems we study may be viewed as taking a more fine-grained approach to traditional influence maximization problems, and correspondingly exhibit non-trivial behavior even in the unbudgeted version where packets are unlimited—including tractability that depends on the probability parameters of the independent cascade dynamics. On the other hand, we obtain NP-hardness results for the budgeted version more analogous to previous work. Several natural heuristics are shown to induce substantial reductions of bias in simulations with well-known social network datasets.

Next, given an undirected graph representing similarities between a set of items and an additive measure evaluating them, we treat the position of a special subset of items in an ordinal ranking through a collection of problems in which items may be combined if they are similar. The objective for these problems is to either maximize or minimize the absolute or relative rank of the special subset, with a meta-goal of assessing the robustness of the rank, even in the presence of a well-defined criterion. We classify the computational complexity of all four problems, mostly finding worst-case hardness, then find exact and approximate solutions to special cases and variants of the problems. These structured cases are inspired by numerous real-world examples and may be used to assess commonly cited facts across disparate domains, as we demonstrate for sources of greenhouse gas emissions that contribute to climate change. We then adapt our framework to require subsets of comparable measure, converting the above ranking problems into redistricting problems, with analogues to the former's grid variants. Emphasis is placed on worst-case deviations from proportional representation and an urban-rural model in which gerrymandering ability is curtailed.

CHAPTER 1

Introduction

The invention of digital computers—and the subsequent emergence of the Internet—has augmented both the transmission and analysis of information within various contexts. Such technology enables rapid interactions between brands and consumers while expanding the social networks in which the latter shares content and discusses ideas. At the same time, the portable availability of even static information arising from expertise and regulated by the government may be aggregated in ways that manipulate and even disenfranchise individuals, regardless of their participation in online spaces. The transfer of information is based on relationships and the aggregation of distinct elements relies on a degree of similarity between them. In this thesis, we explore these dual themes through three projects. The first project concerns the exchange of information seeded by a company. The second project thinks about concept aggregation through the lens of ranking, which is recast in the third project to study redistricting, including the issue of gerrymandering.

All of these scenarios may be modeled with graphs. By representing discrete entities as vertices (nodes), and their connections as edges, graphs constitute a highly flexible framework for modeling these notions. Graphs may be undirected or directed, with the latter representing an asymmetry in the relationship between the two vertices in an edge, and we consider both kinds as dictated by the application. A key structural aspect of an undirected graph is its connected components, which are subgraphs in which all vertices are connected by paths of edges. These subgraphs partition the original graph, and if only one exists, the graph is connected. Directed graphs enjoy an analogous notion of strongly connected components, requiring that the connecting paths respect the directionality of their constituent edges. A cycle is a path that starts and ends at the same vertex with no other vertex repetitions. A directed acyclic graph admits a topological order on its vertices, where edges only point in one direction along which all the vertices may be arranged. A vertex contraction is an operation that allows us to abstract out the difference between two

vertices, effectively merging them while preserving their edges to other vertices. Performing successive vertex contractions within each strongly connected component of a directed graph eventually yields a directed acyclic graph where each component is represented by a single vertex. Numerous special classes of graphs that simplify some of these aspects have been studied. Complete graphs contain an edge between every pair of vertices, an extremity on the gradient of connectivity that sometimes aids in understanding arbitrary graphs. More generally, a regular graph involves each vertex in the same number of edges, and a highly symmetric subtype is circulant graphs. These notions will allow us better understand our application areas of interest, often in surprising ways.

At the same time, to analyze the models built upon these graphs and understand potential outcomes given real-world incentives—we turn to combinatorial optimization, which seeks to find an optimal outcome (or solution) from a finite set of possibilities. As the size of a graph approaches that of a realistic network, merely understanding its contents becomes unmanageable, let alone the exponentially many ways they may interact. Combinatorial optimization provides a more systematic approach to handle this complexity, classifying it by analogy with well-known problems and treating it via polynomial-time algorithms that employ the concepts described in the preceding paragraph. The easiest kind of algorithm to study is greedy algorithms, which exploits a local notion of optimality (e.g., minimizing the cost of one decision) present in some task to optimize it in its entirety (e.g., minimizing the total cost of the task). Other paradigms exist when these procedures are too myopic, most notably dynamic programming, a technique that builds a solution by considering different possibilities and how they relate in an optimal solution. Many computational problems of interest, including some in this thesis, are NP-hard, dashing hopes of finding an algorithm that is both optimal and efficient. Nonetheless, partial resolutions to this quandary exist, including compromise on optimality or efficiency. The former entails approximation algorithms that give a solution that is close to optimal in a provable, proportional sense, while the latter permits exhaustive search only within a (hopefully small) set of decisions mediated by a parameter, giving rise to fixed-parameter tractability. Alternatively, we may direct our efforts to special cases of our problems that align with probable details of the model (e.g., restricting to one of the structured classes of graphs outlined above). This thesis uses all three approaches as needed.

We now preview the three projects alluded to above.

Chapter 2: Influence Maximization with Signed Information

Influence maximization refers to a line of combinatorial optimization problems that seek to spread an idea throughout a social network as far as possible. The canonical example is a company attempting to advertise a product to select consumers who may then influence their contacts with their adoption of it. While some work considers the harmful effect this exposure may have for users or the company, information is usually considered uniform. In Chapter 2, we introduce a more fine-grained influence maximization model that still relies on a directed graph to represent the flow of information among individuals. We treat information as a signed quantity, utilizing packets of $+1$ or -1, where the latter may be interpreted as a negative nudge (e.g., a system compatibility check for a user whose computer will not run a software). This allows us to model the nuanced ways individuals update their granular, pre-existing beliefs based on incoming data for certain non-polarized propositions. Our updating process is justified via Bayes' rule and nonetheless captures the psychological principle that an increasing number of examples in favor of one viewpoint has diminishing returns on a person's core beliefs. By analyzing the network structure, we demonstrate that deterministic belief-updating can be solved efficiently: for acyclic graphs, we use forward substitution, and for directed graphs with cycles, we perform vertex contractions on the pairs of vertices within each strongly connected component. Conversely, we prove that the budgeted version of this problem and its stochastic variants—which closely mirror the noisy, constrained reality of internet advertising and social media feeds—are NP-hard. To address this intractability, we test natural heuristic algorithms on well-known social network datasets, including Gnutella and an Irvine community network. Our simulations demonstrate that substantial reductions in network bias can be achieved without requiring an impractically large number of information packets. Stochastic regimes corresponding to specific deterministic dynamics are also exhibited.

Chapter 3: Ranking with Partitioning

In the Information Age, online platforms routinely summarize complex data by ranking items; examples include lists of cities by affordability or television series by watch time. However, these rankings may be sensitive to the underlying extent to which similar prospective entities are aggregated into just one. In this chapter, we formalize this vulnerability using undirected graphs, where inductively defined walks establish valid structural partitions. We introduce optimization problems aimed at maximizing or minimizing the absolute or relative rank of a specific subset to test the robustness of ordinal metrics. While the general problems are NP-hard, we uncover polynomial-time solutions and approximation algorithms

by restricting the graph structure. For example, the linear component case perfectly models chronologically ordered television episodes grouped into seasons. We demonstrate the real-world implications of these algorithms by analyzing Common Reporting Tables (CRT) for greenhouse gas emissions in the United States. Our numerical experiments reveal that a ranking list——such as the relative contribution of cement production versus aviation——is often just as dependent on the chosen grouping as the underlying data.

Chapter 4: Redistricting with Hierarchical Rectangles

Finally, we apply our graph-partitioning framework to the political sphere, addressing the challenge of democratic redistricting. While contiguous redistricting and population balance are standard requirements for electoral maps, they are largely insufficient to prevent gerrymandering. In this chapter, we attempt to curtail this algorithmic manipulation by enforcing geometric rigidity, restricting permissible districts to axis-parallel rectangles and hierarchical cuts on a grid graph. We prove that, unlike traditional contiguous redistricting, the Gerrymandering with Hierarchical Rectangles problem can be solved exactly in polynomial time via a dynamic programming approach. However, we also demonstrate a theoretical duality: while this geometric rigidity severely restricts extreme partisan packing and cracking, it can structurally preclude proportional representation entirely. We illustrate this limitation using a "pinwheel" construction——a minimal example of five rectangles that cannot be obtained via hierarchical partitioning. To further contextualize these findings within real-world constraints, we introduce an urban-rural model characterized by a linear decay of partisan support moving away from an urban center, with hopeful results for constraining gerrymandering.

CHAPTER 2

Influence Maximization with Signed Information

2.1 Introduction

The advent of online spaces has augmented interest in social networks and their analysis through an algorithmic lens, especially given their importance in advertising marketplaces [24] and the spread of (mis)information [14]. Moreover, machine learning algorithms tailor these forces on the level of individual users [31]. A fundamental problem in this area is that of *influence maximization*, in which an algorithm designer seeks to target select individuals in a social network to maximize the total number of people exposed to a product or information [22, 16, 33]. A key insight from this literature is that a relatively small set of early adopters may be able to promulgate information throughout the network by way of cumulative word-of-mouth effects, which is the aim of *viral marketing*. The efficacy of such a set depends on the topology of the network and the dynamics by which information is passed through the system, which are usually stochastic.

The papers [16] and [33] are among the earliest works to treat the phenomenon of viral marketing mathematically. They employ data mining techniques to optimize advertising expenditures across individuals in a social network by considering their *network value*: the expected gains from a customer influencing their contacts. Some work takes a combinatorial optimization approach to such problems, focusing on choosing a subset of "seed" nodes in a social network in order to maximize the expected number of influenced nodes under various activation functions, in particular the independent cascade model [22]. Though most variants of influence maximization are NP-hard, some do admit efficient approximation algorithms, often by exploiting submodularity of the given objective function [22, 10, 15]. While many variants of influence maximization view information as having a uniform and positive effect on those who receive it, often trying to maximize the expected number of people who do, there are settings where this assumption is not suitable. Recently,

[1] considered a set-up in which the algorithm designer incurs a penalty for information reaching those dubious of it; this models the so-called Groupon effect in which consumers less amenable to a product—but who encounter it due to viral marketing (e.g., via Groupon deals)—tend to decrease its average rating online [9]. Subsequently, [15] addressed the observation that negative opinions may have a greater effect than positive ones by considering the possibility that exposure is negative, wherein negative impressions occur randomly independently of the attempt to influence a node and the node itself. Other work approaches influence maximization in the presence of competition, with node activation warped by the preexisting influence of a more established product [10]. At any rate, packets of information (the act of seeding nodes) are considered uniform in each of these works; they may not be customized by the algorithm designer.

In this chapter, we view information as a signed quantity, reflecting that, while knowledge may have intrinsic value, many people's beliefs about a subject exist on a one-dimensional spectrum; facts in favor of a proposition may also be regarded as facts against a diametrically opposed proposition. Likewise, individuals themselves are treated as heterogeneous, each having preexisting beliefs that are granular but discrete in their formulation. We also suppose that, in a social network with a degree of trust, information received from social ties has some influence on one's beliefs, contrastingly so for "positive" or "negative" information with respect to a preexisting belief, but only based on the stated sign of the information. This is consistent with the notion of belief revision—in particular, our model uses a simple form of Bayesian updating—but not, e.g., confirmation bias or polarization, which is modeled by so-called geometric opinion dynamics [21, 19]. In our model, the goal of the algorithm designer is then to influence individuals' beliefs in a social network—possibly in different directions—based on each person's bias value (positive or negative, integer-valued) ascribed to them by the algorithm designer. We note that the assumptions of our model make it ill-suited to studying political beliefs and how they may be tuned by entities of various intentions due to tendencies for people to interpret political information through their preexisting beliefs (cf., [21, 19]); instead, the model is more appropriate for fine-tuned advertising—inducing positive exposure but not to the point of disappointing key consumers who then write bad reviews (nudges away from a product do exist, such as system compatibility checks for online products [18]), clarifying the answers to empirical questions without any apparent political orientation (people generally start with differing beliefs), and adjusting people's mindsets to help them actualize their potential (e.g., helping students choose a career based on their unique interests and skills). The model is flexible in that it can express the heterogeneity of initial beliefs and the desired end beliefs, the latter

salient for the advertising and career examples (not, hopefully, the empirical matter).

In Section 2.2, we provide a formal description of individual beliefs (via Bayes' rule) and the problem of balancing them via packets (of information) under several objectives, which enables us to better understand the modeling capabilities of the two problems studied in the present paper: Const Info Bal, which includes a standard budget constraint characteristic of influence maximization problems, and Info Bal, which omits this constraint but is nonetheless non-trivial due to the effect of information being node-dependent in our model. In Section 2.3, we obtain a polynomial-time algorithm that solves the deterministic case of Info Bal using basic facts from graph theory, linear algebra, and convex optimization, derive an explicit, combinatorial formula for this unique solution with a priori exponentially many terms, and exhibit an example where exponentially many packets are required. In Const Info Bal and the stochastic case of Info Bal, such examples are shown to preclude several natural algorithms from admitting a multiplicative approximation guarantee for these problems, which are NP-hard in general. In the former, the objective function l_1 is easiest to work with due to linearity (hardness is proved for this objective function and then extended to the others by analytic arguments), whereas in the latter the objective function l_2^2 is best understood due to its description in terms of variance (here, less is known about the other objective functions). Nonetheless, our simulation results suggest that the algorithms we consider often attain considerable improvement in bias throughout the network, and that the number of packets they require is not too large in practice.

2.2 The Model

2.2.1 Intuitive Description

Consider a social network consisting of individuals (asymmetrically) connected by whether they might share information with each other upon learning of something. Each individual has their own integer-valued bias representing the net number of observations they have incurred for a certain dichotomized belief vis-à-vis the true state of the world or the belief otherwise desired for the given person, the latter of which expressed by a bias of 0 (that it is equal to the same number, 0, for each individual is just a normalization step). New packets of information are $\{-1, 1\}$-valued, and when received by an individual, they independently send it to their out-neighbors with some probability pertaining to that relationship à la the independent cascade model [22]. Also upon receiving information, an individual updates their belief by adding the packet's value to their preexisting bias.

The goal of the algorithm designer is to choose a set of *seed* individuals a priori, giving each a possibly distinct number of packets (perhaps differently valued) such that the resultant bias values of the people in the network after the information diffuses are as close to 0 as possible (in expectation).

2.2.2 Converting Examples into Beliefs

In order to make this precise, and capture the principle that an increasing number of net examples in favor of one viewpoint has diminishing effects on a person's beliefs (so that decreasing a small positive bias by 1 is more significant than decreasing a large positive bias by 1), we turn to an epistemic interpretation of beliefs that exhibits these hallmarks. That is, we suppose that there are two states of the world, $+$ and $-$, which disproportionately and independently generate information of the form $+1$ and -1, respectively. Namely, $P(+|+1) = P(-|-1) = \theta \in (1/2, 1)$. Then, let $p \in (0, 1)$ represent the true probability that $+$ is the state of the world (e.g., over epistemic uncertainty remaining if one has knowledge of the state-of-the-art empirical study of a matter; in essence, this is what the algorithm designer wants to impart to individuals in the network (under normalization)).

If individuals update their beliefs according to Bayes' rule, with prior belief on $+$ equal to p, then after n more examples of $+1$ than -1 (regardless of the order in which they arrive), their posterior belief on $+$ is equal to

$$\frac{1}{1 + \left(\frac{1-\theta}{\theta}\right)^n \frac{1-p}{p}} =: \mu(n)$$

[6]. In particular, $\mu(n) \to 1$ as $n \to \infty$ and $\mu(n) \to 0$ as $n \to -\infty$, so $\mu(n)$ reflects the diminishing returns to information as one's beliefs become further entrenched (even true of contrarian examples in that regime). Therefore, the algorithm designer may want to simply minimize the L^1 norm of the vector of posterior beliefs after information diffusion minus $(p, \ldots, p)$, where we note that $\mu(0) = p$, the belief with no net examples. However, we also consider simpler objectives that do not explicit invoke values of μ, which appears to make the analysis much easier.

The implication that one's prior belief was p (before seeing any of the examples forming their initial bias score that the algorithm designer observes) is largely without loss of generality, since we can just incorporate a preexisting belief's deviation from p into their initial bias

value (notwithstanding discrepancies of $\frac{1}{2}\max_{n\in\mathbb{Z}}|\mu(n+1)-\mu(n)| \leq \frac{1}{8}\ln\left(\frac{\theta}{1-\theta}\right)$). Likewise, we may set a different target belief p for different individuals by adjusting their initial bias value (again, notwithstanding discrepancies of $\frac{1}{2}\max_{n\in\mathbb{Z}}|\mu(n+1)-\mu(n)| \leq \frac{1}{8}\ln\left(\frac{\theta}{1-\theta}\right)$), though deviations from a bias of 0 may belie the actual distance from the desired belief, since μ is non-linear. As suggested above, this could reflect one's unique skill set and interests as related to potential careers (one may be weighing two distinct careers, or perhaps two broader categories). Also, by choosing p close to 1, receiving more positive examples than negative ones is essentially optimal under the above objective, so the model encapsulates a traditional influence maximization setting while allowing for more features to be modeled, such as a sharp transition in belief after a person-specific certain number of positive examples (by additionally making their initial bias score the additive inverse of this number and taking θ close to 1 (then, $\mu(n) \sim 0$ for $n < 0$ and $\mu(n) \sim 1$ for $n \geq 0$)).

Unfortunately, as alluded to before, this customizability could potentially be vulnerable to malicious algorithm designers, who may wish for individuals to hold false beliefs or even just mutually incompatible beliefs. This would naturally create skepticism, the type this sort of decentralized learning process seeks to avoid, among individuals in the network who uncover the algorithm designer, especially in regards to matters *recast* as political (we still maintain that overtly political beliefs are unlikely to be manipulated in a way described by our model). Even when well-intentioned, drawing examples inconsistently with their distribution under the true state of the world and seeding them accordingly could be construed as subversive in domains beyond advertising. A contemporary example of such an outcome is local opposition to COVID-19 vaccination campaigns in Zambia, with many residents skeptical of the intentions of pharmaceutical companies in the wake of the AIDS epidemic [27]. An extension of this model might make examples more scarce if they are counter the true state of the world, which should be the same for each individual if encoding an actual fact, while on the other hand modern advertising is unlikely to face such limitations.

In essence, the algorithm designer chooses examples to send through parts of the network in order to balance the beliefs of individuals, depending on their preexisting beliefs and (possibly differing) desired beliefs. The assumption that people update their beliefs according to Bayes' rule, in particular based on a prior that (barring approximation error) should already express probabilistic empirical knowledge of the true state of the world, as if by default manifest in general life experience, may not be realistic.

Nonetheless, it does corroborate the cognitively simpler model in which people just think of

the number of examples they have received for and against a belief in their social circles; even professionals with technical training largely fail to apply Bayes' rule in a more precise, quantitative sense, including when it makes a qualitative difference [11]. The algorithm designer could then implement the information sent to seed nodes by sending targeted facts to these people or coordinating with a trusted community figure in the individual's life (e.g., a teacher).

For the analysis in Section 2.3, it will be helpful to note a few additional properties of μ. First, μ is a logistic function that may be expressed as follows:

$$\mu(n) = \frac{1}{1 + e^{-\ln\left(\frac{\theta}{1-\theta}\right)\left(n - \ln\left(\frac{1-p}{p}\right)/\ln\left(\frac{\theta}{1-\theta}\right)\right)}}.$$

This has the well-known derivative expression

$$\mu'(n) = \mu(n)\left(1 - \mu(n)\right)\ln\left(\frac{\theta}{1-\theta}\right) \leq \frac{1}{4}\ln\left(\frac{\theta}{1-\theta}\right).$$

Hence

$$\max_{q\in[0,1],k\in\mathbb{Z}} |q - \mu(k)| = \frac{1}{2}\max_{n\in\mathbb{Z}} |\mu(n+1) - \mu(n)| \leq \frac{1}{8}\ln\left(\frac{\theta}{1-\theta}\right),$$

which was also used above.

2.2.3 Formal Statement

More formally, let $G = (V, E)$ be a directed graph with V the set of nodes and where an edge $(u, v) \in E$ indicates that u shares a $\{-1, 1\}$-valued packet of information with v (self- and multiple-loops are prohibited, but edges may be bidirectional) upon receiving it with probability $p_{(u,v)} \in (0, 1]$ independently of all other nodes and time steps. For each $v \in V$, there is an associated bias value $b_v \in \mathbb{Z}$. When a node receives information, it updates its bias by adding the value of the packet to its preexisting bias value; each packet (i.e., given to a node by the algorithm designer) may only be received by each node once. All packets are first received by a "seed" node—a node selected by the algorithm designer to receive a given packet. For each node $v \in V$, we refer to the parameter b_v specified by the instance as v's initial bias and v's bias after the updates from all time steps as v's resultant bias b'_v.

Definition 2.2.1. *The* Info Bal *problem is, given a directed graph $G = (V, E)$, probabilities $(p_{(u,v)})_{(u,v)\in E}$, bias values $(b_v)_{v\in V}$, and the independent cascade/bias update dynamics described above, to choose distinct packets to seed to each node such that the nodes' expected*

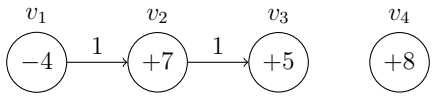


Figure 2.1: An instance of Info Bal with 4 nodes and probability parameter $p_e = 1$ for each edge e. Each node v_i has bias value b_i written inside the corresponding circle. Observe that seeding: 4 packets of $+1$ to v_1; 11 packets of -1 to v_2; 2 packets of $+1$ to v_3; and 8 packets of -1 to v_4 induces a resultant bias vector of $b' = (0, 0, 0, 0)$, hence minimizing each objective function ($l_1(b') = l_2^2(b') = l_\mu(b') = 0$) and solving the corresponding version of Info Bal for this instance. Note that we seed v_1 and v_4 the exact net number of packets required to neutralize their bias since neither has an incoming edge. On the other hand, v_2 must be seeded additional packets since v_1 is neutralized in the opposite direction that v_2 is, whereas v_3 is seeded fewer packets since v_2 is neutralized in the same direction that v_2 is (and the change from v_1's packets is already accounted for in the packets seeded to v_2).

resultant bias values minimize one of the following objective functions of the resultant bias vector $b' = (b'_v)_{v \in V}$*:* $l_1(b') = E[\sum_{v \in V} |b'_v|]$ *(expected* l^1 *norm),* $l_2^2(b') = E[\sum_{v \in V} b'^2_v]$ *(expected sum-of-squares), or* $l_\mu(b) = E[\sum_{v \in V} |\mu(b'_v) - \mu(0)|]$*, where* $\mu(n) = \frac{1}{1+\left(\frac{1-\theta}{\theta}\right)^n \frac{1-p}{p}}$*,* $n \in \mathbb{Z}$ *for given* $\theta \in (1/2, 1)$ *and* $p \in (0, 1)$ *(expected* l^1 *distance between actual beliefs and true state of the world).*

The Info Bal moniker thus refers to three closely related computational problems, with context determining which one (i.e., which objective function) is used. Figure 2.1 gives an example where the solutions under the three objective functions coincide, which does not hold in general for graphs with cycles (Lemma 2.3.4).

Definition 2.2.2. *The* Const Info Bal *problem has the same set-up as the* Info Bal *problem, except that an instance thereof has a budget parameter* $k \in \mathbb{N}$ *meaning that at most* k *distinct packets may be used to seed nodes.*

As above, this definition delineates three computational problems.

Both of these are optimization problems, although they have corresponding decision versions in which a target $B \in \mathbb{N}$ is specified, and the problem is to decide whether there exists a choice of distinct packets to seed to each node such that the resultant objective function value is at most B.

Because the effects of packets are additive and time-invariant, a choice of distinct packets is characterized by the number of $+1$ and -1 packets seeded to each node, and we may analyze the transmission of packets in whichever order is most convenient.

2.3 Results

Unlike other influence maximization problems, which dichotomize the notion of node activation, Info Bal seeks an exact degree of exposure to an idea. This difference allows for a more direct application of techniques from linear algebra and graph theory—Info Bal admits a polynomial-time algorithm in the deterministic case for each of l_1, l_2^2, and l_μ. However, Const Info Bal presents tradeoffs similar to those encountered in traditional influence maximization settings; it is NP-hard with respect to l_1, l_2^2, and l_μ, even in the deterministic case. Moreover, this gradient paradigm means that a relatively large number of packets is required compared to previous models.

2.3.1 The Deterministic Case of Info Bal

We first consider the special case of Info Bal characterized by deterministic dynamics; namely, that $p_{(u,v)} = 1$ for all $(u, v) \in E$. In this case, whenever a node receives information, it travels to all other nodes to which there exists a path (of directed edges). Therefore, if a node u has paths from nodes $v_1, v_2 \ldots, v_{k_u}$, and x_v represents the number of $+1$ packets minus the number of -1 packets seeded to node v, then the change in node u's bias is given by

$$x_{v_1} + \cdots + x_{v_{k_u}} + x_u.$$

If node u has initial bias b_u, then we minimize the summand for node u of the objective function exactly when

$$x_{v_1} + \cdots + x_{v_{k_u}} + x_u = -b_u.$$

This equation admits integral solutions (countably many, in fact) due to the integrality of b_u, leading to the summand for node u of the objective function to be 0. Likewise, the minimum of the objective function that is the smallest possible (a priori), 0, is achieved exactly when the system of equations

$$\left(x_{v_1} + \cdots + x_{v_{k_u}} + x_u = -b_u\right)_{u \in V}. \tag{2.1}$$

admits an integral solution.

We give a characterization of directed graphs for which such a solution (an objective value of 0) is attainable, and obtain a greedy algorithm for attaining the minimum in all cases. We proceed by first considering directed acyclic graphs, then by reducing the problem for arbitrary directed graphs to this case.

Directed Acyclic Graphs. Recall that a directed graph is acyclic if and only if it can be topologically ordered, meaning that we can rename the vertices $V = \{v_1, \ldots, v_n\}$ such that $(v_i, v_j) \in E \implies i < j$. Thus, if G is acyclic, System (2.1) takes the form

$$\left(1_{\{v_1 \to v_j\}} x_{v_1} + \cdots + 1_{\{v_{j-1} \to v_j\}} x_{v_{j-1}} + x_{v_j} = -b_{v_j}\right)_{j \in [n]}, \tag{2.2}$$

where $i \to j$ denotes that there exists a *path* from i to j in the ambient graph.

Forward substitution implies that (2.2) admits a unique solution:

$$\left(x_{v_j} = -b_{v_j} - \sum_{i=1}^{j-1} 1_{\{v_i \to v_j\}} x_{v_i}\right)_{j \in [n]}. \tag{2.3}$$

Note that the b_{v_j} are integral, so (by induction) the x_{v_j} are, too. Thus, the best possible minimum for the objective function of 0 is achieved exactly by giving packets of values with a sum of x_{v_j} for each node v_j. The solution has an intuitive explanation: seed each node the right amount of information to offset their initial bias and any information they have received upstream; since edges only "flow" in one direction, the nodes may be processed sequentially without backtracking.

We can make this more explicit by expressing Solution (2.3) iteratively; that is, inductively substituting the expressions for $x_{v_1}, \ldots, x_{v_{j-1}}$ into the equation for x_{v_j} yields the self-contained expression

$$\left(x_{v_j} = -b_{v_j} + \sum_{i=1}^{j-1} \sum_{k=2}^{j-i+1} (-1)^k \sum_{i=i_1 < \cdots < i_k = j} 1_{\{v_{i_1} \to v_{i_2}\}} \cdots 1_{\{v_{i_{k-1}} \to v_{i_k}\}} b_{v_i}\right)_{j \in [n]}. \tag{2.4}$$

This in an inclusion-exclusion formula that counts, for each $i < j$, the net number of times that the bias b_{v_i} is offset by a node (that receives information from v_i, even if transitively) via information that is seeded to it and ultimately passed on to v_j.

Unfortunately, Solution (2.4) suggests that there is little recourse for efficient parallel computation of the x_{v_j} nor good guarantees on the number of packets required by the optimal solution.

Theorem 2.3.1. *For any $d \in \mathbb{N}$, there exists a family of directed graphs with the number of vertices tending to infinity such that for $j = \Omega(n)$, $|x_{v_j}| = \Omega(n^d)$. Moreover, there exists a family of directed graphs with the number of vertices tending to infinity such that for $j = \Omega(n)$, $|x_{v_j}| = 2^{\Omega(n)}$.*

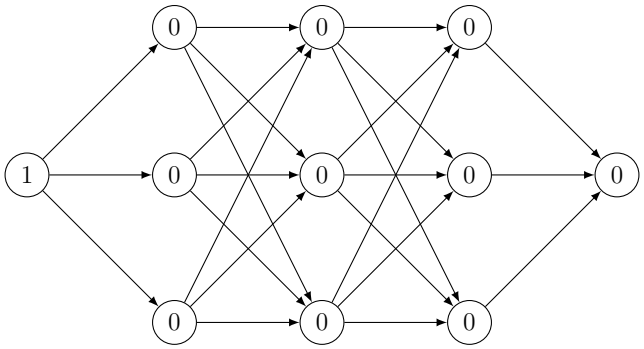


Figure 2.2: A template graph for a potential member of each family of graphs described in the proof of Theorem 2.3.1, with each node's initial bias value written inside the corresponding circle. For this to be a member of the first family (for $d = 2$), the rightmost vertex and edges incident to it are deleted (in which case $n = 10$); for the second family (for $n = 20$), the rightmost vertex is replaced with 10 vertices adjacent to the vertices in the last layer. To generate additional graphs in the first family satisfying the first part of Theorem 2.3.1, the size of each layer is incremented. To generate additional graphs in the second family satisfying the second part of Theorem 2.3.1, the number of layers is incremented.

Proof. To see the first part of the theorem, we construct a graph with an initial vertex v_1 and $d + 1$ layers of $\frac{n-1}{d+1}$ vertices (hence the size of each layer is linear in the number of vertices). The layers are ordered $1, \ldots, d + 1$ such that an edge exists from v_1 to every vertex in layer 1 and from any vertex in a layer s to any vertex in layer $s + 1$; these are the only edges in the graph. Let $b_{v_1} = \Omega(1)$. Also, b_{v_j} for v_j in layer r is $O(n^{r-2})$.

For the second part of the theorem, construct a graph with start and end vertices v_1 and $v_{n/2}, \ldots, v_n$, $\frac{n/2-1}{3}$ layers as in the previous family but with 3 vertices each (hence the number of layers is linear in the number of vertices) and where the nodes in layer $\frac{n/2-1}{3}$ each form an edge with each of $v_{n/2}, \ldots, v_n$. Note that we let $n/2 = 1 \pmod 3$ (since we just need a constant fraction of the vertices after all the layers, this assumption is essentially without loss of generality; there is no cyclic delicacy at play, and the same is true of the above construction). Let $b_{v_1} = \Omega(1)$. Also, b_{v_j} for v_j in layer r is $O(n2^r)$.

The graph constructions are modeled by Figure 2.2. Both parts of the theorem now follow by direct application of (2.4) and the counting techniques used in the proof of Theorem 2.3.2. □

The latter example, at least, is in an asymptotic sense the worst case with respect to packet expenditure.

Theorem 2.3.2. *In Equation (2.4), x_{v_j} is $O(2^j \max_{i \in [n]} |b_{v_i}|)$ uniformly in j, n.*

Proof. In (2.4), the innermost sum has summands of magnitude at most $\max_{i\in[n]} |b_{v_i}|$, so it remains to estimate how many summands there are taken over i and k. Observe that for fixed k, there are at most $\binom{j-i+1}{k}$ summands, so given $0 \leq k \leq j-i+1$, there are at most 2^{j-i+1} summands over k by a standard combinatorial argument. Finally, summing over i, there are at most

$$\sum_{l=1}^{j} 2^l \leq 2^{j+1}$$

summands of the innermost sum. □

These results, while suggesting that using (2.4) directly is not efficient, reveal that forward substitution à la (2.3) can be. Therefore, we can at least ascertain the efficiency of the solution to Info Bal somewhat efficiently. We conclude with a formal statement of this observation.

Theorem 2.3.3. *Iterative application of Equation (2.3) solves the* Info Bal *problem, in the case where the graph is acyclic, in time* $O(n^3 + n^2 \log \max_{i\in[n]} |b_{v_i}|)$.

Proof. That the choice of packets prescribed by (2.3) solves the corresponding instance of Info Bal follows directly from the formula's derivation via (2.1) and (2.2). As for the time complexity, it takes $O(nm)$ time to perform depth-first search with backtracking to determine the connectivity of each pair of nodes and compute a topological ordering. Moreover, Theorem 2.3.2 established that each x_{v_j} is $O(2^j \max_{i\in[n]} |b_{v_i}|)$, so x_{v_j} may be computed by adding $O(j)$ summands with at most $O(j + \log \max_{i\in[n]} |b_{v_i}|)$ digits, incurring $O(j^2 + j \log \max_{i\in[n]} |b_{v_i}|)$ time. In total, the algorithm takes time $O(n^3 + n^2 \log \max_{i\in[n]} |b_{v_i}|)$. □

However, the social network as an acyclic graph, in which no two people mutually exchange information—much less form communities with reciprocity—is quite unrealistic. Fortunately, the solution to Info Bal in the general case follows nicely from the acyclic case, with enhanced implications for the time complexity.

Directed Graphs with Cycles. The other case is that G contains at least one cycle; then, G decomposes into strongly connected components. To solve Info Bal, we perform repeated vertex contractions on the pairs of vertices within each connected component, yielding a directed graph $G' = (V', E')$ where the nodes correspond to the strongly connected components of G and for all $i, j \in V'$, $(i, j) \in E'$ exactly when there exists an edge from a node in component i to a node in component j in G. It follows that G' is acyclic, since a cycle would imply that G contains a cycle spanning multiple connected components, contradicting that strong connectivity be an equivalence relation.

Therefore, this case reduces to the acyclic case as follows. Let $\{1, \dots, C\}$ be the strongly connected components in G; denote the nodes in component $j \in [C]$ by $v_{j,1}, \dots, v_{j,l_j}$ and set $h_j := x_{v_{j,1}} + \cdots + x_{v_{j,l_j}}$ Then System (2.1) becomes

$$\left(1_{\{(1,j)\in E'\}} h_1 + \cdots + 1_{\{(j-1,j)\in E'\}} h_{j-1} + h_j = -b_{v_{j,s}}\right)_{(j\in[C], s\in[l_j])}, \tag{2.5}$$

since all packets received by one node in a strongly connected component i are shared with all remaining nodes in the component, and ultimately all nodes in connected components j such that $i \to j$ in G'. If for all components each constituent node has the same bias, i.e., $-b_{v_{j,s}} = -b_{v_{j,1}}$ for all $j \in [C], s \in [l_j]$, then applying forward substitution as in (2.3) implies that System (2.5) in the *sums* h_j (representing the net value of packets seeded to the nodes in connected component j) admits a unique solution:

$$\left(h_j = -b_{v_{j,1}} - \sum_{i=1}^{j-1} 1_{\{(i,j)\in E'\}} h_i\right)_{j\in[C]}. \tag{2.6}$$

As before, the h_j are integral and hence can be implemented with a number of packets given by the magnitude of the solution (most efficiently). Likewise, Solution (2.6) may also be expressed iteratively:

$$\left(h_j = -b_{v_{j,1}} + \sum_{i=1}^{j-1} \sum_{k=2}^{j-i+1} (-1)^k \sum_{i=i_1<\cdots<i_k=j} 1_{\{i_1\to i_2\}} \cdots 1_{\{i_{k-1}\to i_k\}} b_{v_{i,1}}\right)_{j\in[C]}, \tag{2.7}$$

with connectivity interpreted with respect to G' instead of G.

If instead there exists a connected component j and a pair of nodes therein having different bias values, then by replacing $-b_{v_{j,1}}$ by a constant c_j for each such component j in the above system, there still exists a unique solution (in the sums h_j) such that the bias values in component j shifts by c_j (and become 0 in the remaining components).

In other words, replacing System (2.5) with

$$\left(1_{\{(1,j)\in E'\}}(h_1 + \cdots + 1_{\{(j-1,j)\in E'\}} h_{j-1} + h_j = c_j\right)_{(j\in[C], s\in[l_j])} \tag{2.8}$$

preserves a unique, integral solution

$$\left(h_j = c_j - \sum_{i=1}^{j-1} 1_{\{(i,j)\in E'\}} h_i\right)_{j\in[C]}. \tag{2.9}$$

expressed iteratively as

$$\left(h_j = c_j + \sum_{i=1}^{j-1} \sum_{k=2}^{j-i+1} (-1)^k \sum_{i=i_1<\cdots<i_k=j} 1_{\{i_1 \to i_2\}} \cdots 1_{\{i_{k-1} \to i_k\}} (-c_i) \right)_{j \in [C]}. \tag{2.10}$$

Lemma 2.3.4. *Solution (2.9) solves the* Info Bal *problem if and only if: c_j is a nearest integer of $\frac{-b_{v_{j,1}} + \cdots + -b_{v_{j,l_j}}}{l_j}$, with respect to l_2^2; and c_j is a nearest integer of the set of medians of $\{-b_{v_{j,1}}, \ldots, -b_{v_{j,l_j}}\}$ minimizing l_1 among such integers, with respect to l_1. Computing the $\{c_j\}_{j \in [C]}$ according to these assignments thus takes time $O(n \log \max_{i \in [n]} |b_{v_i}| \log \log \max_{i \in [n]} |b_{v_i}|)$ for the l_2^2 objective function and time $O(n)$ for the l_1 objective function.*

Proof. By construction, Solution (2.9) adds c_j to the bias values of the nodes in component j of G' for each j, whereas by System (2.8), any choice of packets merely adds some $y_j \in \mathbb{N}$ to the bias values of the nodes in component j of G' for each j.

Therefore, minimizing l_2^2 is equivalent to minimizing

$$(b_{v_{j,1}} + y_j)^2 + \cdots + (b_{v_{j,l_j}} + y_j)^2$$

in $y_j \in \mathbb{Z}$ (for each j). Differentiating with respect to $y_j \in \mathbb{R}$ reveals the only critical point is

$$y_j = \frac{-b_{v_{j,1}} + \cdots + -b_{v_{j,l_j}}}{l_j},$$

whereas the function is a quadratic in y_j with positive leading coefficient. Therefore, the minimum over $y_j \in \mathbb{R}$ occurs at

$$y_j = \frac{-b_{v_{j,1}} + \cdots + -b_{v_{j,l_j}}}{l_j},$$

with the function increasing symmetrically in the distance from this point, so the minimum over $y_j \in \mathbb{Z}$ occurs at the nearest integer(s) of $\frac{-b_{v_{j,1}} + \cdots + -b_{v_{j,l_j}}}{l_j}$.

Likewise, minimizing l_1 is equivalent to minimizing

$$|b_{v_{j,1}} + y_j| + \cdots + |b_{v_{j,l_j}} + y_j|$$

in $y_j \in \mathbb{Z}$. As is well-known, the function in $y_j \in \mathbb{R}$ is minimized exactly by the medians of

$$\{-b_{v_{j,1}}, \ldots, -b_{v_{j,l_j}}\}.$$

The standard calculus argument for this fact implies that the function is strictly increasing in the distance from the set of medians to the left and to the right (separately). As a result, the minimum over $y_j \in \mathbb{Z}$ occurs at the integral medians of $\{-b_{v_{j,1}}, \ldots, -b_{v_{j,l_j}}\}$ if one exists, and otherwise the greatest integer less than every median of $\{-b_{v_{j,1}}, \ldots, -b_{v_{j,l_j}}\}$ or the least integer greater than every median of $\{-b_{v_{j,1}}, \ldots, -b_{v_{j,l_j}}\}$ (which of these is the case can be determined numerically, or, using a simple piecewise linear function, analytically). $\square$

In essence, the strongly connected components form communities that update their beliefs by the same increments in relative synchronization due to the ties present between all of their nodes. In the case where nodes' beliefs vary within a connected component, the objective function cannot assume a minimum value of 0. While simpler objective functions of the resultant bias vector like l_1 and l_2^2 admit clean solutions in terms of the c_j—the value that brings the median (interpreted as any convex combination of the two middle entries if n is even) and mean bias value in j closest to 0, respectively—l_μ is a more intricate function of the bias values.

This parallelism with the acyclic case allows for immediate extension of Theorem 2.3.1, Theorem 2.3.2, and Theorem 2.3.3. In order to illustrate that these extensions appear to fortify the time complexity and packet expenditure guarantees of their original analogs, we first reason about the number C of strongly connected components that G is likely to have.

Lemma 2.3.5. *For each $n \in \mathbb{N}$, let P_n be a probability measure on graph $(\{v_1, \ldots, v_n\}, E)$. If, for each n, $P_n(\{(v_i, v_j) \in E\}) > \frac{1.001 \log n}{n}$ uniformly over all $(v_i, v_j) : i \neq j$, and the events $\{(v_i, v_j) \in E\}$ are mutually independently, then almost surely C is $O(1)$ as $n \to \infty$.*

The claim follows from using a coupling argument with the Erdős-Renyi random graph model [3].

This means that, if in social networks each node a priori has an independent baseline probability of sharing information with any other given node inversely proportional to the size of the network but times a factor increasing slowly in n (such that in expectation, there are about $\log n$ such nodes, perhaps reflecting a minimum desired level of socialization in small networks but social "collisions" in busy networks), then the number of strongly connected components in constant in the size of the network with probability 1. Under less stringent assumptions, it is responsible to expect that the number of strongly connected components is $O(\log n)$ as $n \to \infty$ instead—in other words, assuming that $C \leq D \log n$ holds for D a numerical constant independent of n.

Focusing on this more general, latter case, we may now extend the aforementioned theorems.

Theorem 2.3.6. *In Equation (2.10), h_j is $O(2^j \max_{i\in[n]} |c_i|)$ uniformly in j, n. Therefore, if $C = O(\log n)$ as $n \to \infty$, then h_j is $O(n^{dj} \max_{i\in[n]} |b_{v_i}|)$ for $d > 0$ uniform.*

Theorem 2.3.7. *Iterative application of Equation (2.6), with the c_j chosen as specified above, solves the* Info Bal *problem in time $O(C^3 + C^2 \log \max_{i\in[n]} |b_{v_i}|)$. Therefore, if $C = O(\log n)$ as $n \to \infty$, then the time complexity is $O((\log n)^3 + (\log n)^2 \log \max_{i\in[n]} |b_{v_i}|)$.*

2.3.2 The Stochastic Case of Info Bal

We now consider the more general case of Info Bal that experiences stochastic dynamics: $p_{(u,v)} \neq 1$ for some $(u, v) \in E$. This case introduces several additional subtleties.

First, the resultant bias values are now random variables unlikely to be integers. Pertinently, it is #P-hard in general to even calculate the probability that a given packet seeded to u is received by v [32], and many of these events are dependent when v is varied due to overlapping paths. (Fortunately, such events are mutually independent when the node u is distinct for each; distinct packets are transmitted independent.) Nonetheless, we may employ approximation results to estimate these probabilities.

Moreover, note that for any (random variable) b'_v, the identity

$$E[b_v'^2] = E[b'_v]^2 + \mathrm{Var}[b'_v]$$

holds. This equality gives hope that we may replace System (2.1) with one in expectation, as long as the uncertainty incurred by using the requisite number of packets and the rounding from integrality requirements is not too large. Namely, by linearity of expectation, the system

$$\Big(P(x_{v_1} \to x_u)x_{v_1} + \cdots + P(x_{v_{k_u}} \to x_u)x_{v_{k_u}} + x_u = -b_u\Big)_{u\in V}, \tag{2.11}$$

where $P(a \to b)$ is the probability that a packet seeded to node a is (eventually) transmitted to node b, has the property that any solution to the system would result in $E[b'_u] = 0$ for all $u \in V$, albeit not a priori bounding $E[b_u'^2]$ or $E[|b'_u|]$. One subtlety arising from this system is that packets given to nodes within a strongly connected component are no longer interchangeable, as was true for the deterministic case of Info Bal. One approach is to choose an ordering of the nodes within each strongly connected component and otherwise adopt the ordering scheme for the acyclic case: $v_i \to v_j \implies i < j$ (for v_i and v_j not in

the same connected component); adapting the derivation of the deterministic, acyclic case Solution (2.3), we see that the solution would be

$$\left(x_{v_j} = -b_{v_j} - \sum_{i=1}^{j-1} P(v_i \to v_j) x_{v_i} \right)_{j \in [n]}. \tag{2.12}$$

Whether this approach is feasible with respect to our objective functions may depend on p; as $p \to 0, 1$ within a graph, all the above variance terms tend to 0 (uniformly, since any connectivity event that happens with positive probability happens with probability tending to 1 with, say, $p^n \to 0, 1$), so (given near integer increments) the optimization objective is essentially to minimize the $E[b'_v]^2$, which is done by using the deterministic solution of Info Bal as in Theorem 2.3.7 (modulo the vanishing uncertainty from earlier nodes).

On the other hand, if the number of nodes is large compared to $|p - 1/2|$, this may not be the case, e.g., a node connecting to c others with biases 1 and 0 would subject the others to variance sum $cp(1-p) > 1$ for c sufficiently large, hence the optimal solution being to seed no packets to any node.

This suggests that as p moves away from $1/2$, deterministic dynamics weigh more heavily, whereas as n increases, stochastic dynamics become more important. At any rate, we want to mediate between these two factors.

First, let us exhibit an example where the solutions for Info Bal in the deterministic case and the stochastic case diverge sharply, with the solution for Const Info Bal transitioning sharply between these two solutions at a cutoff value of k.

Lemma 2.3.8. *There exists a directed graph $G = (V, E)$, bias values $(b_v)_{v \in V}$, and probabilities of transmission $(p_e)_{e \in E}$ such that the deterministic case of* Info Bal *(i.e., where we ignore the probability parameters) is solved only by seeding packets to each node, the deterministic case of* Const Info Bal *is solved by the solution for the deterministic case or by seeding no packets to any nodes (depending on k), and the stochastic case of* Info Bal *(with respect to l_2^2) is solved only by seeding no packets to any nodes.*

Proof. Construct a graph with an initial vertex v_1, an end vertex v_n, and $d \geq 2$ layers of 3 vertices (hence the size of each layer is linear in the number of vertices). The layers are ordered $1, \ldots, d$ such that an edge exists from v_1 to every vertex in layer 1, from any vertex in a layer s to any vertex in layer $s+1$, and from any vertex in layer d to v_n; these are the only edges in the graph. Set $b_{v_1} = 1$, $b_{v_j} = 0$, $j \neq 1$. Figure 2.2 precisely shows such a

graph for $n = 11$ (in general, graphs in the family will have $\frac{n-2}{3}$ layers). Let each edge have probability of transmission $\frac{1}{2}$. The deterministic case of Info Bal has a unique solution given by (2.3), where we find that $x_{v_1} = -1$; in layer r, the nodes v_r have

$$x_{v_r} = (-2)^{r-1}$$

and

$$x_{v_{n/2}} = \cdots = x_{v_n} = (-2)^{\frac{n/2-1}{3}}.$$

On the other hand, since l_2^2 may only assume integral values and $l_2^2(b) = 1$, the solution to Const Info Bal may decrease l_2^2 only if k is equal to the sum of packets delineated by the x_v; otherwise, $l_2^2(b) = 1$ is the minimum value of l_2^2. As for the stochastic case of Info Bal, note that if a packet of -1 is given to v_1, l_2^2 consists of variance terms of at least

$$3(1/2)(1 - 1/2) + 3(1/4)(1 - 1/4) > 1$$

since the probability each node in layer 1 receives the packet is $\frac{1}{2}$ whereas in layer 2 the probability is between $\frac{1}{4}$ (monotonicity) and $\frac{3}{4}$ (union bound), so the solution does not seed packets to any nodes. □

On the other hand, the next result implies that the stochastic case of Info Bal subsumes the deterministic case of Info Bal in some sense, with the network topology determined by the choice of the probability parameters.

Theorem 2.3.9. *For any directed graph $G = (V, E)$ and bias values $(b_v)_{v \in V}$, there exists a probability threshold $0 < p_- < 1$ such that if $p_e < p_-$ for all $e \in E$, then the solution to the corresponding instance of* Info Bal *with respect to l_2^2 is unique and is given by the solution to the deterministic instance of* Info Bal *obtained by discarding each edge.*

Proof. Our approach is to compare the optimal solution in the deterministic instance (obtained by discarding all edges) to any other choice of seed packets in the stochastic instance.

In the deterministic instance, the unique optimal solution is to seed exactly $-b_v$ packets to each node v, resulting in bias values b'_v. Let $M = \sum_{v \in V} |b_v|$ be the total number of packets seeded. In the stochastic instance, let $p_{\max} = \max_{e \in E} p_e$. The maximum probability that any specific packet successfully traverses the network to reach a node other than its seed is bounded by $1 - (1 - p_{\max})^{n-1}$.

For each node v, its expected resultant bias under the optimal deterministic solution is $E[b'_v] = b_v + (-b_v) + E_v = E_v$, where E_v is the expected net value of packets arriving from other nodes. We bound this by $|E_v| \leq M(1-(1-p_{\max})^{n-1})$. Because packet arrivals are independent, the variance of the bias at v is bounded by the sum of the variances of the M independent packet arrivals: $\mathrm{Var}[b'_v] \leq M(1-(1-p_{\max})^{n-1})$.

The objective function is thus bounded by:

$$l_2^2(b'_v) = \sum_{v \in V} (E[b'_v]^2 + \mathrm{Var}[b'_v]) \leq n\Big[M^2(1-(1-p_{\max})^{n-1})^2 + M(1-(1-p_{\max})^{n-1})\Big].$$

To ensure this solution remains uniquely optimal in the stochastic case, we will require $l_2^2(b'_v) < \frac{7}{16}$. Since $M \geq 1$ and $n \geq 2$, restricting $M(1-(1-p_{\max})^{n-1}) \leq \frac{7}{32n}$ guarantees that this term is less than 1, which allows us to simplify the bound on the quadratic expression:

$$n\Big[M^2(1-(1-p_{\max})^{n-1})^2+M(1-(1-p_{\max})^{n-1})\Big] \leq 2nM(1-(1-p_{\max})^{n-1}) < 2n\left(\frac{7}{32n}\right) = \frac{7}{16}.$$

This condition is satisfied exactly when $1-(1-p_{\max})^{n-1} < \frac{7}{32nM}$.

To find a sufficient condition for $p_{\max}$, we apply Bernoulli's inequality, which states that $(1-p_{\max})^{n-1} \geq 1-(n-1)p_{\max}$. Substituting this into our probability bound yields:

$$1-(1-p_{\max})^{n-1} \leq 1-\Big(1-(n-1)p_{\max}\Big) = (n-1)p_{\max}.$$

Therefore, to guarantee our requirement is met, it is sufficient to restrict $(n-1)p_{\max} < \frac{7}{32nM}$.

Now consider any alternate solution S' where the seeded packets s_v differ from $-b_v$ for at least one node v. Let $d = b_v + x_v$. Since b_v and x_v are integers and $x_v \neq -b_v$, we must have $|d| \geq 1$.

For this node v, the resultant bias is $b'_v = d + \sum_j I_{j,v}$, where $I_{j,v} \in \{-1, 0, 1\}$ is the random contribution of the j-th packet seeded elsewhere. Let $E_v = \sum_j E[I_{j,v}]$, so that $E[b'_v] = k + E_v$.

By independence, $\mathrm{Var}[b'_v] = \sum_j \mathrm{Var}[I_{j,v}]$. Let $q_j = P(|I_{j,v}| = 1)$. By our restriction above, no packet has more than a $\frac{1}{2}$ probability of reaching v, meaning $q_j \leq \frac{1}{2}$. Thus, the variance contributed by each packet is $\mathrm{Var}[I_{j,v}] = q_j(1-q_j) \geq \frac{1}{2}q_j \geq \frac{1}{2}|E[I_{j,v}]|$. Summing this over all packets gives $\mathrm{Var}[b'_v] \geq \frac{1}{2}\sum_j |E[I_{j,v}]| \geq \frac{1}{2}|E_v|$.

We now lower bound node v's contribution to the objective under S':

$$E[b_v'^2] = E[b_v']^2 + \mathrm{Var}[b_v'] \geq (d + E_v)^2 + \frac{1}{2}|E_v|.$$

Since $|d| \geq 1$, the minimum of the continuous function $f(E_v) = (d + E_v)^2 + \frac{1}{2}|E_v|$ occurs when E_v has the opposite sign as d and magnitude $|E_v| = |d| - \frac{1}{4}$. Substituting this into the expression yields the absolute minimum for $E[b_v'^2]$:

$$\left(|d| - \left(|d| - \frac{1}{4}\right)\right)^2 + \frac{1}{2}\left(|d| - \frac{1}{4}\right) = \frac{1}{16} + \frac{|d|}{2} - \frac{1}{8} = \frac{|d|}{2} - \frac{1}{16}.$$

Since $|d| \geq 1$, this minimum value is bounded below by $\frac{1}{2} - \frac{1}{16} = \frac{7}{16}$. Therefore, the total objective for the alternate solution S' satisfies $l_2^2(S') \geq \frac{7}{16}$.

Because we established that $l_2^2(b_v') < \frac{7}{16}$ is mathematically guaranteed whenever $(n-1)p_{\max} < \frac{7}{32nM}$, we can define our threshold as the lower bound:

$$p_- := \frac{7}{32n(n-1)\sum_{i=1}^{n}|b_{v_i}|}.$$

For any $p_{\max} < p_-$, the deterministic solution S^* strictly minimizes the objective, making it the unique optimal solution. The result is tautological for $n = 1$. □

Unfortunately, it is not straightforward to prove an analogous statement for an upper probability threshold p_+. The main obstacle is that, since packets are likely to be transmitted far beyond their seed node if $p_e > p_+ \sim 1$ for all $e \in E$, it becomes harder to estimate the number of packets an optimal solution may a priori use. Additionally, it is still the case that variance terms are smaller than rounding terms in the above sense, which makes a packet-by-packet analysis—namely, one that considers the net effect of a generic packet differing from the deterministic solution and argues it is negative—difficult.

2.3.3 Hardness of Const Info Bal and the Stochastic Case of Info Bal

We argue that Const Info Bal is NP-hard with respect to all three objective functions, even in the deterministic case.

Theorem 2.3.10. *The decision version of* Const Info Bal *with respect to* l_1 *is* NP*-hard.*

Proof. We adapt the standard reduction for influence maximization from Set Cover, as used in [22]. In that case, the use of set cover reflects the binary nature of covering an

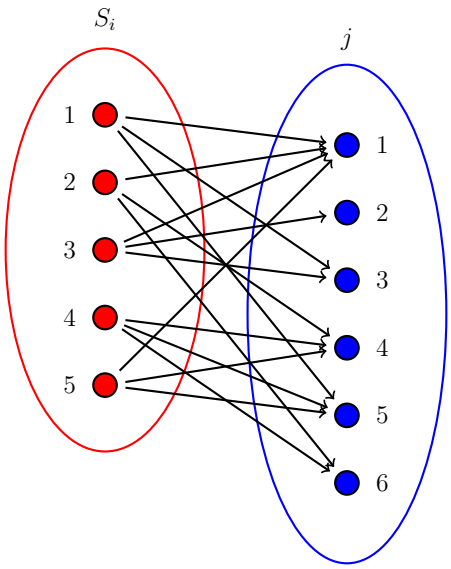


Figure 2.3: The reduction from X3C to Const Info Bal in Theorem 2.3.10, depicted with $m = 5$ and $n = 6$ ($k = 2$). Each node on the left has bias 0, and each node on the right has bias -1.

element/influencing a node. For our problem at hand, it behooves us to reduce from a problem that additionally reflects the extent to which an element is covered, i.e., a node is influenced. We also need to perform a quantitative analysis of the solution. Therefore, we reduce from exact cover by 3-sets, denoted X3C.

Given a (non-trivial) instance of X3C consisting of:

- universe $[n]$, where $n = 3k$ for some $k \in \mathbb{N}$; and
- subsets $S_1, \ldots, S_m \subset [n]$ such that $|S_i| = 3$ for all i,

we construct an instance of the decision version of Const Info Bal with budget k and target k. The graph G consists of nodes

$$V(G) = \{S_1, \ldots, S_m, 1, \ldots, n\},$$

with associated bias values $b_{S_i} = 0$ for all $i \in [m]$ and $b_j = -1$ for all $j \in [n]$; its edge set is

$$E(G) = \{(S_i, j) : j \in S_i\},$$

where each edge has probability of transmission 1. The reduction is illustrated in Figure 2.3. If the X3C instance has a solution, then giving a packet of $+1$ in the Const Info Bal

instance to the node corresponding to each set S_i in the cover results in a bias vector l_1 norm of k (since k subsets are selected and exactly one of them contains each element in $[n]$, hence those nodes having bias 0).

If instead the Const Info Bal instance has a solution, note that

$$l_1(b') - l_1(b) \leq -2k.$$

Observe that the l_1 norm of the bias vector decreases by at most 1 for each of the x packets seeded to elements (they have out-degree of 0) and by at most $3 - 1 = 2$ for each of the at most $k - x$ packets seeded to subsets (since each such packet increases the magnitude of the subset node's bias by 1 (unless packets of both $+1$ and -1 are seeded to that subset node, but then there is no net effect on bias from those 2 packets yet necessarily a remaining element node with bias -1 as each packet reaches at most 3, so this is non-optimal) and decreases that of element nodes' biases by at most 3 in total).

Let $c_1 \leq 1$ and $c_2 \leq 2$ denote the mean respective decreases for each and $k' \leq k$ the number of packets seeded in the optimal solution. But the inequality

$$2(k - x) + 1x \geq c_2(k' - x) + c_1 x \geq 2k$$

holds only if $x \leq 0$ and $k' = k$, so the solution to Const Info Bal must seed k packets to subset nodes. This incurs a l_1 bias of k on the subset nodes, so the bias of the element nodes must decrease by $3k$, which can only happen if each receives exactly one packet; namely, the subsets that are seeded a packet exactly cover the elements. □

Corollary 2.3.11. *The decision version of* Const Info Bal *with respect to* l_2^2 *is* NP*-hard.*

Proof. We argue that the same reduction may be used as in the proof of Theorem 2.3.10. Indeed, the above solution to the Const Info Bal instance corresponding to an arbitrary X3C instance solution consists of bias values of magnitude at most 1, hence the same under l_1 and l_2^2 and thereby constituting a solution to the Const Info Bal instance with respect to l_2^2 as well.

As for any solution to the Const Info Bal instance with respect to l_2^2, this objective function only increases under other bias vectors relative to l_1 while maintaining the value for entries of magnitude 1 or 0, so the solution is of the form obtained in the proof of Theorem 2.3.10. As a result, it likewise corresponds to a solution of the X3C instance. □

Corollary 2.3.12. *The decision version of* Const Info Bal *with respect to* l_μ *is* NP*-hard.*

Proof. We use the same reduction as in the proof of Theorem 2.3.10, except we additionally set $p = \frac{1}{2}$ and $\theta > \frac{1}{2}$ sufficiently close to $\frac{1}{2}$, to be determined later; lastly, the target is changed to

$$\left(\theta - \frac{1}{2}\right) k.$$

The main idea is to ensure that μ is approximately linear when restricted to the relevant part of its domain for a solution to the instance of Const Info Bal, then apply the argument for Theorem 2.3.10.

As in the proof of Theorem 2.3.10, a solution to the X3C instance yields a solution to the Const Info Bal instance with

$$l_\mu(b') = (\theta - \frac{1}{2})k$$

(since an example of $+1$ gives the subset nodes a belief of θ, compared to the true belief $\frac{1}{2}$).

For the reverse direction, note that the worst possible value of $l_\mu(b')$ with k packets is at most $4k$, with it being at most $k+1$ in magnitude for the corresponding summand for any individual node. Let θ be sufficiently small so that

$$\frac{\mu'(n+x)}{\mu'(x)}$$

is within ϵ of 1 for $|n| \leq k+1, x \in [0,1]$ (this can be done since $\mu(k+2)$ can be made arbitrarily close to $\frac{1}{2}$ by making $\ln(\frac{\theta}{1-\theta})$ sufficiently close to 0, and these are polynomial-time calculations), so then all such values are within ϵ proportionally. Thus the summand of $l_\mu(b')$ for a node with $|m| \leq k+1$ examples is between

$$|m|(1-\epsilon)\left(\theta - \frac{1}{2}\right)$$

and

$$|m|(1+\epsilon)\left(\theta - \frac{1}{2}\right),$$

as can be seen by integrating the derivative μ'. Now divide each summand of l_μ by $\theta - \frac{1}{2}$; thus there exists a solution with $l_\mu(b') = k$. Moreover, the summand for a node with $|m| \leq k+1$ packets is between $|m|(1-\epsilon)$ and $|m|(1+\epsilon)$. Let $\epsilon = \frac{1}{8k}$ so that $l_\mu(b')$ is always within $\frac{1}{2}$ of the value for the corresponding l_1 solution.

Thus the corresponding l_1 solution has l_1 bias at least $k - \frac{1}{2}$ and at most $k + \frac{1}{2}$. But in the l_1 case analyzed in the proof of Theorem 2.3.10, a l_1 bias score of k necessarily corresponded to an exact set cover, and the increments in l_1 were integral, so the solution to the CONST INFO BAL instance still converts into an exact set cover for the corresponding X3C instance. □

In fact, INFO BAL is also NP-hard with respect to l_2^2—in the stochastic case (Section 2.3.1 established that the deterministic case admits a polynomial-time algorithm).

Theorem 2.3.13. *The decision version of* INFO BAL *with respect to* l_2^2 *is* NP*-hard.*

Proof. As in the proof of Theorem 2.3.10, we reduce from X3C. This time, we modify the reduction to have auxiliary nodes j' for each $j \in [n]$ with added edges $(j, j') \in E$. Each edge has probability of transmission $p := \frac{1}{2}$. The subset nodes have bias -1; element nodes, -1; auxiliary nodes, 0. The target is

$$m + \frac{5}{12}n.$$

The reduction is shown in Figure 2.4.

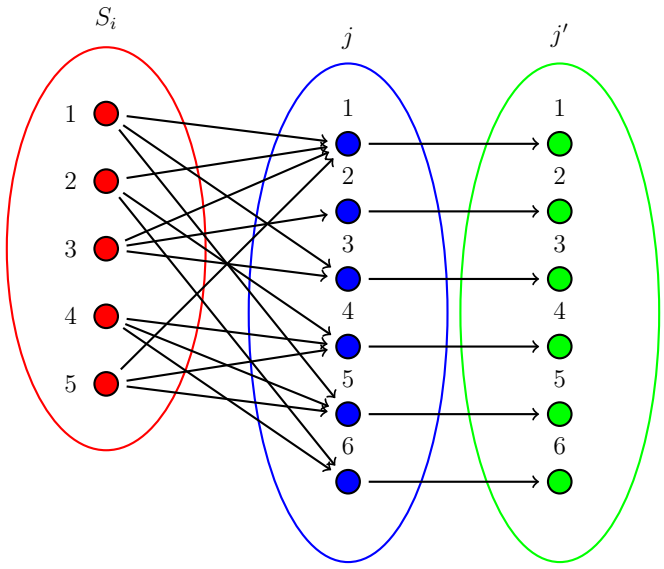


Figure 2.4: The reduction from X3C to INFO BAL in Theorem 2.3.13, depicted with $m = 5$ and $n = 6$ ($k = 2$). Each node on the left has bias -1, each node in the center has bias -1, and each node on the right has bias 0.

The reason we now need non-zero bias on the subset nodes and the presence of auxiliary nodes with zero bias is to incentivize seeding a limited number of packets to subset nodes and disincentivize seeding packets to element nodes, respectively, but both effects are achieved predominately due to the variance experienced from seeding packets to a node

with positive out-degree.

As a result, a solution to the instance of INFO BAL will not seed packets to any element nodes, just one packet to each subset node constituting an exact cover of the X3C instance (there, the increase in variance is offset by the decrease in squared expected bias of the relevant subset and element nodes, since such nodes start with non-zero bias of the same sign).

First suppose that the instance of X3C has a solution. For any subset node S_i and its corresponding three element nodes j with their three auxiliary nodes j', we have $E[b_{S_i}^2] = (-1)^2 = 1$, $E[b_j^2] = (-1)^2 = 1$, and $E[b_{j'}^2] = 0^2 = 0$. Upon seeding a packet of $+1$ to S_i, we obtain $E[b_{S_i}'^2] = E[b'_{S_i}]^2 + \mathrm{Var}[b_{S_i}] = 0^2 + 0 = 0$, $E[(b'_j)^2] = E[b'_j]^2 + \mathrm{Var}[b'_j] = (-1/2)^2 + (1/2)(1 - 1/2) = 1/2$, and $E[(b'_{j'})^2] = E[b'_{j'}]^2 + \mathrm{Var}[b'_{j'}] = (1/4)^2 + (1/4)(1 - 1/4) = 1/4$. Taking the difference between the first three terms and the last three terms, seeding each subset node in the exact cover a packet of $+1$ in the INFO BAL instance results in l_2^2 decreasing by

$$\frac{n}{3}\Big[1 + 3(3/4) - 3(1/4) - 3(1/16) - 3(3/16)\Big] = \frac{7}{12}n,$$

whereas

$$l_2^2(b) = m + n,$$

so

$$l_2^2(b') = m + \frac{5}{12}n$$

as desired.

Now suppose instead that the instance of INFO BAL has a solution. We argue that the number of times l_2^2 may decrease after seeding a packet is limited, and no combination of seeds may result in l_2^2 decreasing by $\frac{7}{12}n$ unless only the subset nodes in an exact cover for the X3C instance are seeded (a packet of $+1$). A simple calculation implies that only packets of $+1$ may be optimally seeded to subset nodes.

First, recall from above that seeding a packet of $+1$ to a subset node that has not received any packets (where this status also holds for its element nodes and the associated auxiliary nodes) decreases l_2^2 by $7/4$ (type I packet). If instead exactly one of the element nodes (and its associated auxiliary node) has received a packet of $+1$, we analogously find that l_2^2

decreases by

$$1 + 2(3/4) + 1(1/4) - 3(1/4) - 1(3/16) - 2(1/16) - 3(3/16) = 9/8$$

(type II packet). If instead exactly two of the element nodes (and their associated auxiliary nodes) have received a packet of $+1$, l_2^2 decreases by

$$1 + 1(3/4) + 2(1/4) - 3(1/4) - 2(3/16) - 1(1/16) - 3(3/16) = 1/2$$

(type III packet). On the other hand, if all three of the element nodes (and their associated auxiliary nodes) have received a packet of $+1$, l_2^2 increases by at least

$$3(1/4) + 3(1/4) - 3(1/16) + 3(3/16) - 1 - 3(1/4) = 1/8,$$

and if some of the element nodes have instead received two packets of $+1$ in any of the foregoing, l_2^2 may only increase.

Similarly, if a subset node has already received a packet of $+1$, seeding it another packet necessarily increases l_2^2 (the above decreases in l_2^2 are strictly less than 2); the same is true of seeding element or auxiliary nodes under any circumstances. With the former, it incurs variance of $1/4$ for doing so, whereas its squared expected bias of $1/4$ or 0 is not helped (unless it has received 4 packets from subset nodes, but then the additional variance from 3 extra packets offsets this by at least $\frac{1}{4}$ vis-à-vis receiving 1 packet, so the packet types above achieve the minimum value of l_2^2 given the baseline of 1 packet for those that have already received packets), and that of its auxiliary node decreases by at most $1/4$. Hence for the latter, its squared expected bias likewise may not be improved by seeding it integral packets.

Let x, y, and z represent the number of type I, II, and III packets used in the Info Bal solution, respectively. Each type I, II and, III packet requires 3, 2, and 1 element nodes, respectively, that have not received any other packets, so

$$3x + 2y + z \leq n.$$

But type II packets decrease l_2^2 by more than 2 times the amount type III packets do, and type I packets decrease l_2^2 by more than $3/2$ times the amount type II packets do. As a result, l_2^2 is uniquely minimized by $x = \frac{n}{3}$, the only case in which its target is met, and this

choice of seed nodes corresponds to an exact cover for the X3C instance. □

It is worth noting that the quantity $l_2^2(b')$ for any given random bias values b' induced by an optimal solution of seed packets is continuous in p, and is pointwise equicontinuous for $p \in (0,1)$ with respect to the family $(l_2^2(b'))_{b' \in \mathbb{Z}^n}$ given the limit on seed packets that court variance terms, which is locally bounded for $p \in (0,1)$. So, because none of the inequalities used in the above proof were sharp, it follows that for p sufficiently close to $\frac{1}{2}$, the above instance of Info Bal is still NP-hard. That is, the reduction works for moderate values of p and does not hinge on delicacies like expected bias values being integral.

The analysis for this reduction does not quite work with respect to l_1; determining infeasible packet types for the reverse direction relies on the convexity of the squaring function, which for one packet type renders negative the sign of the change in the objective function (since b'_v and b'^2_v do not coincide for some v). Nonetheless, we expect that Info Bal is NP-hard with respect to l_1 and l_μ for moderate probability of transmission parameters, as is the case with respect to l_2^2, but modifying the above reduction is not immediate per the commentary after its statement.

Lastly, we note that even a 2-approximation algorithm for Info Bal would just correspond to, e.g., an exact cover for exactly half the underlying set in the above reductions, of which there are many halves combinatorially, so the reductions in this section do not appear to yield inapproximability results.

The analysis in Lemma 2.3.8, however, allows us to argue that a few natural heuristic approaches to Const Info Bal and the stochastic case of Info Bal admit no multiplicative approximation guarantee.

First, consider the heuristic algorithm of truncating the optimal solution using M packets down to the prescribed budget k (we restrict to the interesting case of $k < M$). Namely, given Solution (2.9) with h_j net packets seeded to the jth strongly connected component, the algorithm instead seeds

$$\left\lfloor \frac{k}{M} h_j \right\rfloor \tag{2.13}$$

net packets to the jth strongly connected component. The intuition behind this algorithm is that scaling the optimal solution by $\frac{k}{M}$ would result in each connected component finishing with $1 - \frac{k}{M}$ of its original bias value (the value c_j computed based on which objective function

is used) by linearity, so this is an interpolation technique that avoids the computationally hard task of assessing the relative value of each packet seeded to a node.

Corollary 2.3.14. *The truncation algorithm for* Const Info Bal *that uses Equation (2.13) does not approximate the optimal solution within any multiplicative factor for* l_1, l_2^2, *or* l_μ.

Proof. Consider the graph shown in Figure 2.2 (with the naming conventions from Lemma 2.3.8). Given $k < M$, we have $x'_u := \left\lfloor \frac{k}{M} x_u \right\rfloor < \frac{k}{M} x_u$ for all $u \in V$ (since in the deterministic solution, $x_u > 0$ for all $u \in V$). In particular, the algorithm sets $x'_{v_1} = x'_{v_2} = x'_{v_3} = x'_{v_4} = 0$. Assuming $k \geq \frac{M}{2^{(n-2)/3}}$, there exists $v_i \in V$ of minimum index i such that $x_{v_i} \neq 0$, so then $b_{v_1} \geq 1$ *and* $b'_{v_i} \geq 1$ (since $b_{v_i} = 0$ and no preceding nodes are given packets that may neutralize the bias), meaning the algorithm increases the value of each objective function relative to the optimal solution of not using any packets. □

We also consider heuristics for the stochastic case of Info Bal. One approach is intuitive: convert to the deterministic case by preserving an edge if and only if its probability of transmission p is greater than $\frac{1}{2}$, then use Solution (2.7) for the deterministic case. The idea is that if this condition is met, each packet sent along an edge is more likely than not to reach the incident node and affect its bias, whereas otherwise the packet likely will not have this effect (we make the inequality $p > \frac{1}{2}$ strict because sending packets also results in bias terms that increase l_2^2).

Corollary 2.3.15. *The above algorithm for the stochastic case of* Info Bal *that reduces the edge set and then uses the deterministic case Solution (2.7) does not approximate the optimal solution within any multiplicative factor for* l_2^2.

Proof. Consider the graph shown in Figure 2.2. There, we found that the variance term in l_2^2 is strictly greater than 1 for probability of transmission $p = \frac{1}{2}$. By continuity of that expression in p, there exists $\epsilon > 0$ such that if $p_e \in (1/2 - \epsilon, 1/2 + \epsilon)$ for all edges e incident to v_1, the variance term from any packet is still greater than 1 but yet this algorithm would prescribe sending packets to at least v_1 (this is true regardless of which probability parameters are greater than $\frac{1}{2}$), whereas the bias, all from v_1, may only be reduced by 1. □

Similarly, for the algorithm that uses the "in expectation" Solution (2.12) (with non-integral packets prescriptions x_u replaced with $[x_u]$), v_1 is seeded a packet and other vertices are too (depending on the layer it is in and the probability parameters) in the graph from Figure 2.2. Thus the algorithm does not admit a multiplicative approximation guarantee.

2.3.4 Simulation Outcomes

In order to address the NP-hardness of Const Info Bal even in the deterministic case, we consider the heuristic algorithm of truncating the optimal solution using M packets down to the prescribed budget k, as described by Equation (2.13). The algorithm is tested using the well-known Gnutella social network dataset, a peer-to-peer file-sharing system [25], and the Irvine social network, a "Facebook-like" online community for University of California, Irvine students [29]. In every trial, each node's bias is sampled independently and uniformly from $\{-100, -99, \ldots, -1, 0, 1, \ldots, 99, 100\}$. We perform 70 trials in total, in which a mean $\overline{M}$ of 294945.6 and 30561.48571 packets are used for the optimal Solution (2.6) in the respective networks. More specifically, for 7 different values of k, roughly evenly spaced between 0 and $\overline{M}$, we perform 10 trials; the empirical ratios $\frac{M}{k}$ vary (mostly due to k) in whether they are close to a rational number expressible with small numerator and denominator, which is relevant to ascertaining if the optimal solution is likely to be scaled down with (much) rounding error. Despite this variation, the observed (mean) decay in the objective function l_2^2 is remarkably close to quadratic in the value of k for both datasets, interpolating between the observed objective function value under no packets and under the optimal solution with a constant of proportionally of $\left(1 - \frac{k}{\overline{M}}\right)^2$ (and there is little variation in the objective function within the 10-sets of trials). This suggests that in practice, the algorithm may often enjoy such an approximation guarantee, though since the bias values are drawn from the same distribution for each node, this may belie the importance of selecting the most impactful packets arising from the convexity of l_2^2—as opposed to obliviously scaling down the optimal solution in a way that is symmetric with respect to the nodes in the network. However, this does suggest that an analogous linear decay would be observed for l_1. The outcomes are tabulated in Table 2.1.

Gnutella (8717 vertices)		Irvine (1900 vertices)	
Packets (k)	**Objective (l_2^2)**	**Packets (k)**	**Objective (l_2^2)**
872	2.93×10^7	190	6.42×10^6
$2,179$	2.94×10^7	475	6.44×10^6
$4,358$	2.94×10^7	950	6.42×10^6
$8,717$	2.86×10^7	$1,900$	6.38×10^6
$73,861$	2.16×10^7	$7,668$	5.52×10^6
$147,722$	1.59×10^7	$15,337$	4.91×10^6
$221,584$	1.22×10^7	$23,005$	4.51×10^6

Table 2.1: The mean (over 10 trials of Const Info Bal) of observed l_2^2 objective function values for the given dataset using Equation (2.13). Regression analysis of these values suggests quadratic decay with r^2 values of .9997 (Gnutella) and .9994 (Irvine), with regression coefficients all statistically significant at the .001 level.

In order to address the NP-hardness of Info Bal in the stochastic case, we consider the heuristic solution for l_2^2 that replaces System (2.1) with one in expectation, as described by Equation (2.12), and also the edge deletion heuristic from Corollary 2.3.15. For the former, each probability of transmission $P(v_i \to v_j)$ is estimated via 100 Monte Carlo simulations. The algorithm is tested on the Gnutella and Irvine datasets. Simulation results are detailed in Table 2.2, which captures the full spread of outcomes across trials, including several extreme outliers. Each bias value is independently generated from a Cauchy distribution with parameters $x_0 = 0$ (location) and $\gamma = 1$ (scale); the heavy tails may reflect long-range dependencies indicative of how actual opinions evolve over time. As can be seen, the heuristic algorithm generally reduces the value of the objective function l_2^2 from the initial bias values by a large amount, with the effect being more pronounced for smaller probabilities of transmission (.1 vs. .2) and generally in the desired direction (decrease). The algorithm also seems to work better on the Gnutella dataset, perhaps owing to that it only consists of one strongly connected component with more than one node (albeit having 3000 such nodes), rendering the optimization within each strongly connected component (or even which nodes to seed first) less important compared to the Irvine dataset. It should also be noted that, even per node, the squared bias values are still quite high, and similar preliminary results are obtained for Gaussian bias values (whose tails are much lighter), though Jensen's inequality allows us to take the square root to upper bound the observed bias value (in magnitude) per node, which is more modest—often quite small.

Heuristic	**Dataset**	**Probability (p)**	**Change in Objective (l_2^2)**
System in Expectation (Equation 2.12)	Gnutella	0.1	6.29×10^7, 1.99×10^8, 1.08×10^8, 7.82×10^7, 6.64×10^8, 1.21×10^9, 1.20×10^9, 1.01×10^8, 5.90×10^8, 3.67×10^6, 4.57×10^8
	Gnutella	0.2	8.07×10^7, 1.39×10^8, 1.45×10^8, 9.80×10^6, 3.81×10^7, 1.21×10^8, 3.59×10^7, 1.86×10^8, 6.85×10^7, -7.56×10^8, 6.76×10^6
	Irvine	0.1	3.17×10^9, -3.19×10^7, 7.15×10^8, 3.81×10^{10}, 1.16×10^7, -2.28×10^7, -5.19×10^6, 3.66×10^6, -2.29×10^7, 1.91×10^6, 4.19×10^9
	Irvine	0.2	-1.45×10^8, 4.25×10^6, -3.71×10^8, -3.50×10^8, -2.98, 5.10×10^7, -1.37×10^7, -1.92×10^7, -2.90×10^9, -9.19×10^7, -3.86×10^8
Edge Deletion (Corollary 2.3.15)	Gnutella	0.1	7.94×10^7, 4.56×10^7, 1.90×10^{11}, 1.95×10^7, 2.86×10^8, 1.11×10^{10}, 3.19×10^8, 1.45×10^{10}, 5.95×10^7, 3.84×10^7, 2.17×10^{10}, -1.37×10^9, -1.78×10^{10}, -1.46×10^9, -4.83×10^{10}, -7.51×10^9, -1.53×10^{10}
	Gnutella	0.2	6.10×10^7, 8.53×10^7, 8.68×10^9, 5.08×10^7, 1.45×10^7, 8.90×10^6, 7.18×10^9, 1.99×10^8, 3.05×10^8, 3.71×10^8, 1.69×10^9
	Irvine	0.1	4.31×10^8, -3.95×10^7, -9.28×10^7, -8.83×10^7, -7.95×10^7, -5.44×10^9, 4.18×10^6, -1.49×10^9, -9.15×10^7, -3.89×10^6, -8.14×10^8
	Irvine	0.2	-7.63×10^7, -7.16×10^8, -5.91×10^8, -1.92×10^8, -5.71×10^7, -1.19×10^{10}, -1.16, -2.52, -1.50, 1.00×10^4, -1.87×10^9

Table 2.2: Observed changes in the l_2^2 objective function upon applying the given heuristic approach.

CHAPTER 3

Ranking with Partitioning

3.1 Introduction

Rankings play a central role in how information is summarized and communicated in the Information Age. Online platforms routinely rank cities by livability, universities by academic quality, and pieces of media by popularity or quality. Outside of explicitly digital settings, rankings induce metrics that are frequently cited as evidence in policy debates, journalism, and scientific communication.

Despite generally being presented as plain facts in the latter contexts, many such rankings are constructed from entities that admit multiple reasonable levels of aggregation. Cities may be grouped into metropolitan areas or countries, food ingredients into the byproducts formed when following a recipe, and television episodes into seasons or series. These aggregations are typically justified by some notion of similarity, yet allow for many degrees of freedom in general, even under domain-specific constraints. As a consequence, the position of a particular item or collection of items in a ranking may depend not only on the underlying measure being applied, but also on how similar items are combined or kept separate.

This observation raises a basic robustness question: to what extent is the rank of a given item intrinsic to the measure being used, and to what extent can it vary under different groupings by similar items?

In this chapter, we formalize this question using a framework in which items are represented as vertices of a graph encoding similarity, admissible aggregations correspond to connected subsets of vertices, and rankings arise from additive measures evaluated on these subsets. Within this framework, we study optimization problems that minimize or maximize the rank, or a normalized version called rank percentile, of a designated subset, thereby

quantifying the range of rankings consistent with a fixed underlying metric. Even when historical and legal considerations preclude any meaningful possibility of regrouping, say, land into cities, such considerations are often the result of arbitrary but solidified decisions. Studying all possible partitions enables us to assess counterfactual developments, providing one angle to decide which sociological facts are largely a matter of the definitions and boundaries chosen.

This chapter is organized as follows. Section 3.2 presents the general formulation of the optimization problems, formalizing our motivating examples involving cities and episodes. As three of the four problems stated are NP-hard, Section 3.3 introduces simplifications to our model that arise naturally when modeling several partitioning contexts, including the aforementioned episode example, to make these problems more tractable, yielding polynomial-time optimal solutions and approximation algorithms. Section 3.4 adjusts the set-up of our problems as needed to describe related aggregation problems that may not be realized in the framework of Section 3.2, particularly in the presence of hierarchical categories as in ingredient lists. Our results in Section 3.4 also facilitate the application of our methods to study different sources of greenhouse gas emissions in the United States in Section 3.5, revealing that the relative contribution of many sources depends heavily on our choice of aggregation, even though these ordinal ranking are frequently cited without elaboration.

While this chapter focuses on the new combinatorial optimization problems motivated and introduced above, they enjoy several special cases and variants that are equivalent to well-studied set packing and graph problems, enabling us to better understand the computational complexity and (in)approximability of these alterations. Maximizing the number of vertices selected from intersection and overlap graphs, particularly rectangle intersection graphs, has been shown to be computationally challenging; early hardness results were established by Rim and Nakajima [34], and more recent work has extended these limitations to closely related graph classes such as CPG and EPG graphs [12]. Despite these barriers, approximation algorithms have been developed for related problems, including the $(2+\epsilon)$-approximation for maximum independent set of rectangles due to Gálvez et al. [20]. The k-set packing problem, which arises as an extremal case of partitioning in this chapter, is also hard, but has been approached via large-neighborhood local search techniques by Sviridenko and Ward [36]. Structural graph theory further illuminates settings where structure simplifies our problems, as in the characterization of Hamiltonian properties in abelian group graphs by Chen and Quimpo [13].

3.2 The General Case of the Problem

Consider n indivisible elements $V = \{v_1, \ldots, v_n\}$ that have some interpretation as atomic entities in some category, such as a city or an episode of a show. There is a binary relation $E \subset V \times V$ representing whether two entities are similar and may be combined, as in the cities forming a land mass (e.g., a country) or a series of consecutive episodes of a show (e.g., a season). The relation E is reflexive and symmetric. As suggested by this naming, we may identify such a tuple (V, E) with an undirected (simple) graph $G = (V, E)$, using context to distinguish between the interpretations.

Now, we may partition V into subsets $S_1, \ldots, S_c$ taken from a collection of subsets $\mathcal{S} \subset 2^V$. The collection $\mathcal{S}$ is an inductively defined set: each $S \in \mathcal{S}$ is either a singleton or the union of two subsets in $\mathcal{S}$ forming a pair of similar elements (and $\mathcal{S}$ comprises all such subsets). In other words, $\mathcal{S}$ consists of the subsets S that can be built up by starting with an initial element and successively adding elements similar to some preexisting element (note that the initial element may be taken arbitrarily without loss of generality, since E is symmetric and we can backtrack as needed). To see the equivalence, clearly any such subset satisfies the inductive definition, and when taking the union of two sets $S_1 \cup S_2 \in \mathcal{S}$ as above, if they both satisfy this property (tautological for singletons), then we can start with any element in the union, successively add similar elements from that subset, and, after encountering the similar element from the other subset, add the similar elements from that subset. Thus by structural induction, the property holds for $S_1 \cup S_2$. In graph theoretic terms, $\mathcal{S}$ consists of the subsets S that constitute a walk in G (because two elements being similar means there exists an edge between them in G, and we may use backtracking to toggle between the preexisting elements to only include the ones we want).

Equivalently, $S \in \mathcal{S}$ if and only if for every partition $S = A \sqcup B$, $A, B \neq \emptyset$, there exists $a \in A$ and $b \in B$ such that $(a, b) \in E$ (we say such a subset is *connected* in analogy to the topological definition). If $S = \{u_1, \ldots, u_k\}$ is connected, we can successively build $S \in \mathcal{S}$ starting from u_1 and at each step adding an element of S that has not yet been added (apply the definition of connectivity to the preexisting set and its complement within S). On the other hand, if $S \in \mathcal{S}$ and S is not connected, this means two subsets A and B forming S are in different connected components of G, which contradicts that there be a walk in G consisting of the vertices of S.

Lastly, let $\mu : 2^V \to [0, \infty)$ be a finite measure given by $\mu(S) = \sum_{s \in S} \mu(\{s\})$, where

$\mu(S) = 0 \implies S = \emptyset$. Let $S^* \in \mathcal{S} \cup \{\emptyset\}$ be a special subset whose value $\mu(\mathcal{S})$ we wish to compare to the values $\mu(S)$ for subsets $S \in \mathcal{S}$. Consequently, we write $G \backslash S^*$ to denote the graph with vertex set $V \backslash S^*$ and edge set E minus any edges incident to an element of S^*.

The Rank Minimization problem is the following: given a tuple (V, E, μ, S^*) as above, select a partition $(S_1, \ldots, S_c, S^*)$ of V satisfying $S_1, \ldots, S_c \in \mathcal{S}$ (call such a partition *valid*) that minimizes the number of S_i for which $\mu(S_i) > \mu(S^*)$ (we say one plus this number is the *rank* of S^* under a valid partition).

The Rank Maximization problem is similar: given a tuple (V, E, μ, S^*) as above, select a partition $(S_1, \ldots, S_c, S^*)$ of V satisfying $S_1, \ldots, S_c \in \mathcal{S}$ that maximizes the number of S_i for which $\mu(S_i) \geq \mu(S^*)$ (we say one plus this number is the *rank* of S^* under a valid partition).

The definition of rank differs slightly depending on the problem, and any claims in this chapter about the minimum or maximum rank in some context are satisfied for both definitions simultaneously. To avoid manipulation of rank by such tie-breaking precedent and the number or average size of subsets, we also consider percentile versions of these optimization problems.

The Rank Percentile Minimization problem is the following: given a tuple (V, E, μ, S^*) as above, select a partition $(S_1, \ldots, S_c, S^*)$ of V satisfying $S_1, \ldots, S_c \in \mathcal{S}$ that minimizes the fraction

$$\frac{|\{i \in [c] : \mu(S_i) > \mu(S^*)\}| + .5|\{i \in [c] : \mu(S_i) = \mu(S^*)\}| + .5}{c+1}.$$

The Rank Percentile Maximization problem is similar: given a tuple (V, E, μ, S^*) as above, select a partition $(S_1, \ldots, S_c, S^*)$ of V satisfying $S_1, \ldots, S_c \in \mathcal{S}$ that maximizes the fraction

$$\frac{|\{i \in [c] : \mu(S_i) > \mu(S^*)\}| + .5|\{i \in [c] : \mu(S_i) = \mu(S^*)\}| + .5}{c+1}.$$

In either case, we call this fraction the *percentile* of S^* under a valid partition.

The following definitions will also be convenient for our analysis of the problems.

Definition 3.2.1. *We say that a subset $S \subset V$ is large if $\mu(S) > \mu(S^*)$ and $S \in \mathcal{S} \backslash \{S^*\}$, medium if $\mu(S) = \mu(S^*)$ and $S \in \mathcal{S} \backslash \{S^*\}$, and small if $\mu(S) < \mu(S^*)$ and $S \in \mathcal{S} \backslash \{S^*\}$. By identifying $v \in V$ with $\{v\}$, we also say v is large, medium, or small if $\{v\}$ is large, medium, or small, respectively.*

As an example, we may consider the northeast region of the United States commonly re-

ferred to as New England, which is somewhat arbitrarily partitioned into states, at the more granular level of their minor civil divisions, henceforth referred to as municipalities. Letting V be equal to the set of municipalities in New England, E the set of adjacent pairs of municipalities, and μ the population of an input subset of municipalities, the optimization problems allow us to fix just one pre-established contiguous region of interest to people, such as Boston or Massachusetts itself, and study the extent to which its relative population in the New England region could change under an alternate set of (still contiguous) cities or states.

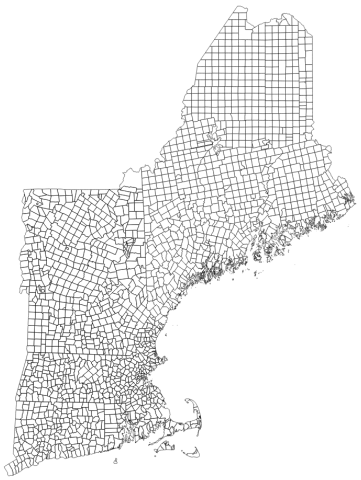


Figure 3.1: A map depicting each municipality in New England by black boundary lines [7, 37]. There are many ways one could have partitioned these municipalities into contiguous states, each way carrying implications for how people may think about the relative standing of its constituent parts under some metric.

To analyze the computational problems stated, we first recall the following standard arithmetic fact.

Lemma 3.2.2. *Let $a, b, c, d \in \mathbb{R}$, where $b, d > 0$. Then* $\min\left(\frac{a}{b}, \frac{c}{d}\right) \leq \frac{a+c}{b+d} \leq \max\left(\frac{a}{b}, \frac{c}{d}\right)$.

Proof. Writing $\frac{a+c}{b+d} = \left(\frac{b}{b+d}\right)\frac{a}{b} + \left(\frac{d}{b+d}\right)\frac{c}{d}$, we see that $\frac{a+c}{b+d}$ is a convex combination of $\frac{a}{b}$ and $\frac{c}{d}$. □

Our main application of Lemma 3.2.2 is in understanding changes to the percentile of a partition upon combining subsets.

Lemma 3.2.3. *Let P_0 be a valid partition of V. There exists a coarser valid partition achieving percentile less than (respectively greater than) that under P_0 if and only if there*

exist l large subsets, m medium subsets, and s small subsets in P_0 whose union is in $\mathcal{S}$, where $l + m + s \geq 2$, such that

$$\frac{l + .5m - 1}{l + m + s - 1}$$

is greater than (respectively less than) the percentile under P_0.

Proof. For the case $l = 0, m = 0, s \geq 2$, $\frac{l+.5m-1}{l+m+s-1} < 0$, which is necessarily less than the percentile of P_0; modifying P_0 so that the s small subsets are combined into one subset, the denominator of the rank decreases whereas the numerator can only increase, so the percentile is greater than that of P_0.

Otherwise, we know that $l \geq 1$ or $m \geq 1$. Therefore, upon combining l large subsets, m medium subsets, and s small subsets into a single subset, the percentile becomes $\frac{l_0-l+.5(m_0-m)+1+.5}{l_0-l+m_0-m+s_0-s+1+1}$. Now by Lemma 3.2.2, the percentile of P_0 (the initial percentile) is a convex combination of this new percentile and the fraction $\frac{l+.5m-1}{l+m+s-1}$, so the new percentile is less than the initial percentile if and only if $\frac{l+.5m-1}{l+m+s-1}$ is greater than the initial percentile, and the new percentile is greater than the initial percentile if and only if $\frac{l+.5m-1}{l+m+s-1}$ is less than the initial percentile. Since any coarser partition of P_0 may be realized by recursively combining subsets of this form, the claim follows by induction: if the new percentile has not decreased after a certain number of steps, then the condition for it to decrease (in terms of available subsets to combine and the required value of $\frac{l+.5m-1}{l+m+s-1}$) only becomes more stringent. The argument for the percentile increasing is symmetric. □

The minimization problems are quite different from each other in that minimizing the rank tends to create relatively few subsets that are larger, which results in a high percentile.

Theorem 3.2.4. Rank Minimization *is polynomial-time solvable.*

Proof. Consider the following greedy algorithm. Set $i = 1$ and mark all nodes in $G \backslash S^*$ as unexplored. While there exists an unexplored $v_+ \in V \backslash S^*$ that is large (i.e., $\mu(\{v_+\}) > \mu(S^*)$), initialize $S_i = \emptyset$, perform breadth-first search in $G \backslash S^*$ starting at v_+. Add each node that becomes explored (traversed by the breadth-first search subroutine) to S_i and increment i. Denote by V_- the set of nodes in $G \backslash S^*$ that remain unexplored upon the algorithm's termination. We claim that the subsets $S_1, \ldots, S_c, \{v_-\}$, where $v_- \in V_-$, that are generated by this procedure minimize the rank of S^*. By the above characterization in terms of walks in G (which includes all walks in $G \backslash S^*$), they are each in $\mathcal{S}$, and by construction they partition $V \backslash S^*$. Observe that the rank of S^* is one plus the number of connected components of $G \backslash S^*$ containing some $v_+ \in V_+$. But subsets spanning multiple connected components are not in $\mathcal{S}$ (nor intersect with S^*), so this is optimal. □

This gives us our first lower bound on the rank of S^*.

Corollary 3.2.5. *The minimum rank of S^* under a valid partition is $1 + C_l$, where C_l is the number of connected components with a singleton that is large.*

Theorem 3.2.6. Rank Percentile Minimization *is* NP*-hard.*

Proof. We show that solving Rank Percentile Minimization, or just determining whether any partition achieves a percentile less than the partition P_0 comprised of S^* and singletons, solves exact cover with 3-sets, X3C. We describe a reduction from an instance of X3C with n nodes and m subsets to an instance of Rank Percentile Minimization.

- Create a node for each subset and element; the only edges are from subsets to the elements contained therein and between each pair of subset nodes
- The subset nodes are small and the element nodes are large (as singleton sets)
- Create isolated large and small nodes (interpreted as singleton sets) as needed so that the percentile of S^* under P_0 is $p_0 := \frac{n-1}{n+n/3-1} - \frac{1}{2}(\frac{n-1}{n+n/3-1} - \frac{n-3-1}{n-3+n/3-1-1})$
- All small nodes are so small that the sum of their values is less than $\mu(S^*)$

We show that there is an exact set cover if and only if there exists a subset such that $\frac{l+.5m-1}{l+m+s-1} > p_0$. By Lemma 3.2.3, this condition is equivalent to the statement that there exists a partition with percentile less than the percentile of P_0, so this completes the proof.

$\implies$: Take the subset to consist of the subset nodes comprising an exact cover (and all of the element nodes), since the subset nodes are all connected and each element node is connected to the subset node covering it in the exact cover. Then $\frac{l+.5m-1}{l+m+s-1} = \frac{n-1}{n+n/3-1} > p_0$.

$\impliedby$: We analyze such a subset and show its subset nodes constitute an exact cover. Isolated nodes cannot be part of the (non-singleton) subset. Similarly, any element node part of the subset must have a corresponding subset node covering it in the subset. For m subset nodes in subset, want to add maximum $3m$ element nodes to the subset, but this is less than p_0 by construction and that each subset-3-element tuple added increases the fraction by Lemma 3.2.2. If instead $m = k$, at least one of the subset nodes only bring 2 new element nodes, but the fraction of $\frac{2}{3}$ decreases the fraction from the baseline given by a 1:3 ratio (true at minimum of the "first" subset node in the subset in some enumeration thereof). □

On the other hand, the maximization problems are similar in that maximizing the rank is generally consistent with maximizing the rank percentile: make as many large subsets as possible.

Theorem 3.2.7. Rank Maximization *and* Rank Percentile Maximization *are* NP*-hard.*

Proof. We reduce from Multiway Number Partitioning. We are given an instance $\langle B, k\rangle$, where B is a multiset of positive integers summing to T and k represents the number of multisets in our partition of B (so $k|T$). We now reduce to an instance of Rank Maximization. Let V be equal to B plus a special element v^* (create multiple vertices for duplicate numbers), $E = V \times V$, and $S^* = \{v^*\}$. Define μ by $\mu(\{v^*\}) = \frac{T}{k}$ and otherwise $\mu(\{v\})$ is equal to the integer in B corresponding to the element v. Now if the instance of Multiway Number Partitioning has a solution, this establishes that the rank of S^* may be at least $k+1$, and if such a rank is achieved, by positivity it must be the case that each of the k subsets with greater value than S^* under μ have value exactly equal to $\frac{T}{k}$. The reduction to Rank Percentile Maximization is similar, except we must perturb μ slightly so that S^* has value strictly smaller than the other subsets (this also establishes hardness of rank maximization when the inequality in the definition of rank is taken to be strict). For this, now let $\mu(S^*) = \frac{T}{k} - \frac{1}{2}$. If the instance of Multiway Number Partitioning has a solution, then a percentile of $\frac{k+.5}{k+1}$ is possible. Now we extend to the other direction. By integrality of B, for k subsets to be greater than S^* under μ, we need each to still have value $\frac{T}{k}$, and this is the maximum number of subsets possible, hence uniquely maximizing the percentile of S^* (since by inspection, the percentile is a convex combination of 1, .5, and 0, with the former having the greatest coefficient when the largest number of subsets have value greater than S^* and all subsets of $V \backslash S^*$ are of this form, making the coefficient of the latter 0). Thus the percentile of $\frac{k+.5}{k+1}$ is only achieved by a solution to the instance of Multiway Number Partitioning. □

Moreover, in a somewhat restricted set of examples, the maximization problems are hard to solve even approximately.

Theorem 3.2.8. *If* P $\neq$ NP*, then for any* $\epsilon > 0$ *there is no polynomial-time* $(\frac{3}{2} - \epsilon)$*-approximation algorithm for* Rank Maximization*.*

Proof. Taking $k = 2$ in Multiway Number Partitioning, we obtain the Partition problem, which is still NP-hard. Perform the same reduction to Rank Maximization as in the proof for Theorem 3.2.7. Now if the Partition instance has a solution, the optimal value for the Rank Maximization instance is 3. Note that for any $\delta > 0$, $(\frac{2}{3} + \delta) \cdot 3 > 2$, meaning that such an approximation algorithm must solve Partition exactly (if the Partition instance does not have a solution, the optimal value for the Rank Maximization instance is less than 3, so an approximation algorithm would also report this). This would imply that P = NP. □

Theorem 3.2.9. *If* P $\neq$ NP*, then for any* $\epsilon > 0$ *there is no polynomial-time* $(\frac{10}{9} - \epsilon)$*-approximation algorithm for* Rank Percentile Maximization*.*

Proof. The proof is similar to that for 3.2.8, except we apply the reduction to Rank Percentile Maximization from Theorem 3.2.8 and need to compute the ratio of the optimal value in both cases. If the Partition instance has a solution, the maximum percentile is $\frac{2.5}{3}$, whereas otherwise it is at most $\frac{1.5}{2}$ (this is the case where one large subset is possible, and we make that the only other subset in the partition; it is always optimal to eliminate a small subset compared to leaving it in the partition, and medium subsets are not possible since $\mu(S^*)$ is not integral). Lastly, we observe that $\frac{1.5/2}{2.5/3} = \frac{9}{10}$. □

A subtlety arises in our analogue between the above maximization problems when medium subsets are present (which act as a hybrid between large subsets and small subsets in the percentile optimization problems), but we may still convert Rank Percentile Maximization to Rank Maximization without too much loss of optimality.

Lemma 3.2.10. *There is a 2-approximation algorithm for* Rank Percentile Maximization *by reducing the problem to multiple instances of* Rank Maximization *in polynomial-time. Moreover, the algorithm is optimal in the case that the maximum percentile is at least .5.*

Proof. The high-level idea is to take the better of the two percentiles obtained by (1) maximizing the number of large subsets in the partition; and (2) maximizing the number of large or medium subsets in the partition. Let l^*, m^*, and s^* denote the number of large, medium, and small subsets in a valid partition achieving maximum percentile p^*, respectively. Since Lemma 3.2.3 implies that combining a small subset with any other subset increases the percentile under any partition, s^* must be equal to the number of connected components of G that have value (as a subset of vertices) less than $\mu(S^*)$ under μ. Now applying Lemma 3.2.2, we can express p^* as $\frac{l^*}{l^*+m^*+s^*+1} + .5 \cdot \frac{m^*+1}{l^*+m^*+s^*+1}$.

First suppose that the first term is larger. Now create an instance of Rank Maximization with input identical to the instance of Rank Percentile Maximization except that the new measure $\tilde{\mu}$ is equal to μ on sets having empty intersection with S^* and at S^* is $\frac{a'}{b_1 b_2 \cdots b_n}$, where $\mu(\{v_1\}) = b_1, \mu(\{v_2\}) = b_2, \ldots, \mu(\{v_n\}) = b_n$ and a' is the smallest integer so that the expression is greater than $\mu(S^*)$ (which can be found by solving $\frac{x}{b_1 b_2 \cdots b_n} = \mu(S^*)$ and rounding up). Observe that maximizing rank here is equivalent to maximizing the number of large subsets in our original instance. Consider the partition returned by solving the Rank Maximization instance, and further minimize the number of small

and medium subsets (with respect to μ) by absorbing them into large subsets within the given connected component of G or, if there are no such subsets in the connected component, leaving it as one subset. Let l_1, m_1, and s_1 denote the number of large, medium, and small subsets in the resultant partition, respectively. By the preceding sentence, we have $l_1 \geq l^*$, $m_1 \leq m^*$, and $s_1 = s^*$, so the resultant partition has percentile $\frac{l_1}{l_1+m_1+s_1+1} + .5 \cdot \frac{m_1+1}{l_1+m_1+s_1+1} \geq \frac{l^*}{l^*+m^*+s^*+1} \geq .5\left(\frac{l^*}{l^*+m^*+s^*+1} + .5 \cdot \frac{m^*+1}{l^*+m^*+s^*+1}\right)$.

If instead the second term is larger, we create an instance of Rank Maximization whose input is completely identical to the instance of Rank Percentile Maximization. Consider the partition returned by solving the Rank Maximization instance, and further minimize the number of small subsets as above. Let l_2, m_2, and s_2 denote the number of large, medium, and small subsets in the resultant partition, respectively. Observe that $l_2 + m_2 \geq l^* + m^*$ and $s_2 = s^*$, so the resultant partition has percentile

$$\begin{aligned}
\frac{l_2}{l_2+m_2+s_2+1} + .5 \cdot \frac{m_2+1}{l_2+m_2+s_2+1} &= .5 \cdot \frac{l_2}{l_2+m_2+s_2+1} + .5 \cdot \frac{l_2+m_2+1}{l_2+m_2+s_2+1} \\
&\geq .5 \cdot \frac{l^*+m^*+1}{l^*+m^*+s^*+1} \\
&\geq .5 \cdot \frac{m^*+1}{l^*+m^*+s^*+1} \\
&\geq .5\left(\frac{l^*}{l^*+m^*+s^*+1} + .5 \cdot \frac{m^*+1}{l^*+m^*+s^*+1}\right).
\end{aligned}$$

Therefore, by returning the partition with the larger percentile of the two partitions constructed above, we guarantee that it has at least $\frac{1}{2}$ of the maximum percentile p^*.

Lastly, if $p^* \geq .5$, then since $l \leq 1, m \geq 0, s \geq 0 \implies \frac{l+.5m-1}{l+m+s-1} \leq .5$, there exists a partition attaining percentile p^* that does not have any medium or small subsets in a connected component that has value greater than $\mu(S^*)$ under μ (since for any partition with percentile at least .5, Lemma 3.2.3 implies combining l large subsets, m medium subsets, and s small subsets in the above ranges may only increase the percentile). By inspection, the first partition constructed above is such a partition, so it will be returned by our algorithm (or another partition of the same percentile). Conversely, if the algorithm returns a partition with percentile at least .5, then it must be the case that $p^* \geq .5$, so our algorithm returns the maximum percentile by our preceding analysis. □

3.3 Algorithms for Special Cases

3.3.1 Complete Graph Case–Bounding Rank and Percentile

The first special case of the problem we consider is where every pair of objects is similar, i.e. G is a complete graph. The *complete graph* case enables us to find (sharp) bounds on the rank and percentile and, correspondingly, may be used to model actors who optimize rank or percentile based on ex-post descriptions of similarity/categorization (instead of actually being constrained by it a priori).

We start with the minimization problems, both of which admit closed-form solutions in this setting.

Corollary 3.3.1. *If G is a complete graph, the minimum rank of S^* under a valid partition is 1 if no vertex is large (as a singleton) and 2 otherwise.*

Proof. The statement is a direct application of Corollary 3.2.5. □

The next statement tells us that, in the salient regime of this special case, the minimum percentile is on the order of $\frac{1}{s_0}$, but may become close to .5 if there is not both a subset that is larger and a subset that is smaller than S^*.

Theorem 3.3.2. *Let l_0, m_0, and s_0 be the number of vertex singletons that are large, medium, and small, respectively. Define $\mathbf{1}_{\{l_0>0\}}$ to be 1 if $l_0 > 0$ and 0 otherwise, with analogous definitions involving m_0 and s_0. If G is a complete graph, the minimum rank of S^* under a valid partition is $\frac{\mathbf{1}_{\{l_0>0\}}+.5}{\mathbf{1}_{\{l_0>0\}}+\mathbf{1}_{\{l_0=0\wedge m_0>0\}}+s_0+1} \leq .5$ if ($l_0 \geq 1$ and $s_0 \geq 1$) or ($m_0 \geq \frac{2s_0+1}{s_0-1}$ and $s_0 \geq 2$), $\frac{.5m_0+.5}{m_0+s_0+1} \leq .5$ if ($l_0 = 0$ or $s_0 \geq 1$) and ($m_0 < \frac{2s_0+1}{s_0-1}$ or $s_0 \leq 1$), and $\frac{.5m_0+1.5}{m_0+2} > .5$ if $l_0 \geq 1$ and $s_0 = 0$.*

Proof. In each case, we exhibit a partition achieving the given percentile and argue that it is the minimum. The partition is formed as follows, starting from all singletons. Combine all large singletons into one subset. Then, if ($l_0 \geq 1$ and $s_0 \geq 1$) or ($l_0 = 0$, $m_0 \geq \frac{2s_0+1}{s_0-1}$, and $s_0 \geq 2$), add all medium singletons into the (possibly empty) subset comprised of all large singletons.

By Lemma 3.2.3, there must not be more than one large subset in the final partition (since combining two of them necessarily decreases the percentile, and we may always do this in a complete graph), so all large vertices must be in the same subset in the partition. Similarly, all small singletons must be subsets in the final partition. Therefore, the only degrees of

freedom we have are combining medium subsets or combining medium subset(s) and the large subset (if it exists). Consider the updated partition we obtain after the first step (aggregating all large singletons). If $l_0 \geq 1$ and $s_0 = 0$, then the updated partition achieves percentile greater than .5, and applying Lemma 3.2.3 with $l \leq 1$ and $s = 0$ reveals that no coarser partition has smaller percentile. If ($l_0 = 0$ or $s_0 \geq 1$) and ($m_0 < \frac{2s_0+1}{s_0-1}$ or $s_0 \leq 1$), then by the first condition the updated percentile is at most .5, and the second is equivalent to $\frac{.5m_0-1}{m_0-1} < \frac{.5m_0+.5}{m_0+s_0+1}$, meaning that combining the medium subsets would increase the percentile per Lemma 3.2.3 (or any subset thereof, since the left-hand side of the inequality is increasing in m_0, and note that there are no large subsets to combine with in this case). Lastly, if ($l_0 \geq 1$ and $s_0 \geq 1$) or ($m_0 \geq \frac{2s_0+1}{s_0-1}$, and $s_0 \geq 2$), then by the first condition the updated percentile is at most .5, and the second is equivalent to $\frac{.5m_0-1}{m_0-1} \geq \frac{.5m_0+.5}{m_0+s_0+1}$, so by Lemma 3.2.3, combining the medium singletons is optimal even if there are no large subsets, and if there is a large subset, it is optimal to add the medium singletons to that since doing so eliminates the maximum number of medium subsets, which is desirable since the updated percentile is at most .5, while not adding any new large subsets, which would increase the percentile. Thus in all cases, the algorithm computes an optimal partition, and by inspection this yields the claimed percentile. □

On the other hand, the maximization problems still have inherent intractability (the inapproximability results, Theorem 3.2.8 and Theorem 3.2.9, were found in this special case), but may now be approximated within a constant factor.

The following definitions will be useful for our approach.

Definition 3.3.3. *An element $v \in V$ intermediate if $\frac{1}{2}\mu(S^*) \leq \mu(\{v\}) < \mu(S^*)$ and tiny if $\mu(\{v\}) < \frac{1}{2}\mu(S^*)$. Given a constant $\frac{1}{2}\mu(S^*) \leq c < \mu(S^*)$, we say an element $v \in V$ is subintermediate if $\mu(S^*) - c \leq \mu(\{v\}) < \frac{1}{2}\mu(S^*)$.*

Theorem 3.3.4. *The complete graph case of* Rank Maximization *has a polynomial-time approximation algorithm with approximation guarantee* $\mathrm{OPT} \leq \frac{3}{2}\mathrm{ALG} + 1 - \frac{1}{2}(l_0 + m_0) - \frac{1}{2}B_0^* - \frac{1}{6}B^*$, *where l_0 and m_0 are the number of vertex singletons that are large and medium, respectively, and B_0^* and B^* are the number of subsets comprised of exactly 2 and 1 intermediate elements in an arbitrary optimal solution, respectively.*

Proof. Two key insights for our algorithm is that (1) a subset needs exactly 2 intermediate elements to become medium or large if no tiny elements are used; and (2) medium or large subsets comprised entirely of tiny elements never overshoot the threshold $\mu(S^*)$ by a factor exceeding $\frac{3}{2}$. We will also need to analyze subsets combining intermediate and tiny

elements, since the tiny elements may exceed the *remainder* of the threshold $\mu(S^*)$ by a factor exceeding $\frac{3}{2}$, which affects how many subsets may be formed from remaining tiny elements. These dynamics motivate Definition 3.3.3.

We now describe the algorithm and then state it formally as Algorithm 1. Suppose there are exactly r intermediate elements $\{v_1, \ldots, v_r\} \in V$ and index V such that $\mu(\{v_1\}) \geq \cdots \geq \mu(\{v_n\})$. Initialize $P = \emptyset$. For each $v \in V$ that is medium or large (as a singleton set), add $\{v\}$ to P. For each $i \in [\lceil \frac{r}{2} \rceil, r]$ that is even, add $\frac{i}{2}$ subsets to P, each containing 2 intermediate elements (and nothing else) from the set $\{v_{r-i+1}, \ldots, v_r\}$. Now initialize a subset $S_j = \{v_j\}$ for each of the $r - i \in [0, \lfloor \frac{r}{2} \rfloor]$ remaining intermediate elements $v_j : 1 \leq j \leq r - i$ and add it to P. Set $c = \mu(\{v_{r-i}\})$ and let s_1 be the number of subintermediate elements of V. For each natural number $x \in [0, s_1]$, add 1 distinct subintermediate element to the x smallest S_j. For the remaining $r - i - x$ largest S_j, add any tiny elements not added to any subset thus far one-by-one to S_j until $\mu(S_j) \geq \mu(S^*)$ (such a subset is called *completed*). Lastly, add any tiny elements not added to any subset thus far one-by-one to an empty subset until it is completed, transitioning to a new empty subset whenever this criterion is met. Add each of these initialized subsets to P. The algorithm returns $\text{argmax}_{i,x} P$.

Algorithm 1 Given an instance $(V = \{v_1, \ldots, v_n\}, \mu, S^*)$ of Rank Maximization in the complete graph case such that $\mu(\{v_1\}) \geq \cdots \geq \mu(\{v_n\})$, compute the maximum rank of S^*

Initialize $P(\cdot, \cdot) \leftarrow \emptyset$
Add each large or medium $v \in V$ to $P(\cdot, \cdot)$ as a singleton
for i in $\left[\lceil \frac{r}{2} \rceil, r\right] : i$ even **do**
  Split $\{v_{r-i+1}, \ldots, v_r\}$ into disjoint pairs and add each pair to $P(i, \cdot)$
  $T_1(i, \cdot,) \leftarrow \{v \in V : v$ is subintermediate relative to $\mu(\{v_{r-i}\})\}$
  $T_1^0(i, \cdot) \leftarrow |T_1(i, \cdot)|$
  for j in $[1, r-i]$ **do**
    $S_j(i, \cdot) \leftarrow \{v_j\}$
  end for
  for $x \in \mathbb{N} : x \leq T_1^0(i, \cdot) \wedge x \leq r - i$ **do**
    for j in $[r - i - x + 1, r - i]$ **do**
      Move $e \in T_1(i, x)$ with smallest μ-measure to $S_j(i, x)$
      $P(i, x) \leftarrow P(i, x) \cup \{S_j(i, x)\}$
    end for
    $R(i, x) \leftarrow T_1(i, x) \cup \{v_{r+T_1^0(i,\cdot)}, \ldots, v_n\}$
    for j in $[1, r - i - x]$ **do**
      while $\mu(S_j(i, x)) < \mu(S^*)$ and **do**
        Move $e \in R(i, x)$ with smallest μ-measure to S_j
      end while
    end for
    while $R(i, x) \neq \emptyset$ **do**
      $S \leftarrow \emptyset$
      while $R(i, x) \neq \emptyset$ and $\mu(S) < \mu(S^*)$ **do**
        Move $e \in R(i, x)$ with smallest μ-measure to S
      end while
      $P \leftarrow P \cup \{S\}$
    end while
  end for
end for
return $\text{argmax}_{i,x} P(i, x)$

More formally, to compare W and W^*, we assume without loss of generality that $\mu(S_1^* \backslash \{v_1^*\}) \leq \cdots \leq \mu(S_{B^*}^* \backslash \{v_{B^*}^*\})$ (for any reversed pair of subsets, we may swap their non-intermediate elements while keeping each complete). Whenever our algorithm adds 1

subintermediate element to a subset, the subset becomes completed; because $x = x^*$, and we choose the smallest (with respect to value under μ) such elements first, we have that each of the summands $\mu(S_{r-i-x+1} \backslash \{v_{r-i-x+1}\}), \ldots, \mu(S_{r-i} \backslash \{v_{r-i}\})$ in W is less than or equal to the summand in W^* of the same index. As for the remaining summands in W, each corresponding subset must have been completed by an element with value less than $\frac{1}{2}(\mu(S^*) - c) \equiv \frac{1}{2}(\mu(S^*) - \mu(v_{\lfloor \frac{B^*}{2} \rfloor}))$, so those summands are at most $\mu(S^*) - \mu(\{v_j\}) + \frac{1}{2}(\mu(S^*) - \mu(v_{\lfloor \frac{B^*}{2} \rfloor}))$, but at least $\mu(S^*) - \mu(\{v_j\})$ (for the largest half) or $\mu(S^*) - \mu(v_{\lfloor \frac{B^*}{2} \rfloor})$ (for the smallest half). Thus

$$W \leq \sum_{j=1}^{\lfloor \frac{B^*}{2} \rfloor} (\mu(S^*) - \mu(\{v_j\})) + \left\lfloor \frac{B^*}{2} \right\rfloor \cdot \frac{1}{2}(\mu(S^*) - \mu(v_{\lfloor \frac{B^*}{2} \rfloor}))$$

and

$$W^* \geq \sum_{j=1}^{\lfloor \frac{B^*}{2} \rfloor} (\mu(S^*) - \mu(\{v_j\})) + \left\lceil \frac{B^*}{2} \right\rceil (\mu(S^*) - \mu(v_{\lfloor \frac{B^*}{2} \rfloor})),$$

so $W \leq W^*$. Since the algorithm completes $S_1, \ldots, S_{\lfloor \frac{B^*}{2} \rfloor}$ and pairs the other $\lceil \frac{B^*}{2} \rceil$ intermediate elements to form $\left\lfloor \frac{\lceil \frac{B^*}{2} \rceil}{2} \right\rfloor$ subsets, $B_0^* + B^* \leq B_0 + \frac{4}{3}B + 1$.

At a high level, this iteration of our algorithm splits B^* into halves, choosing to pair the smallest half of intermediate elements with each other while still completing a subset for each of the remaining largest half of intermediate elements (using tiny elements). This ensures that our algorithm still captures $\frac{2}{3} + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}$ of the subsets from B^*, and classifying tiny elements relative to the value that the smallest intermediate element in our largest half needs to be completed allows us to bound the value of tiny elements used with intermediate elements relative to an optimal algorithm, resulting in similar tiny element values "left over" to be combined for more subsets. The resulting partition, and how it differs from the optimal partition, is depicted in Figure 3.2.

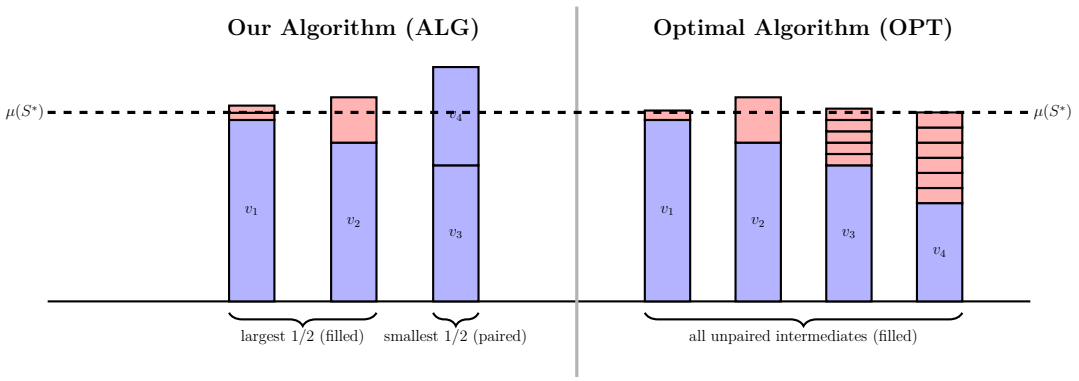


Figure 3.2: A visual representation of Algorithm 1 for unpaired intermediate elements used by an optimal algorithm. Our algorithm completes the largest half of the intermediate elements with tiny elements and pairs the remaining half with each other (among the intermediate elements not paired by the optimal algorithm), guaranteeing they cross the $\mu(S^*)$ threshold without relying on further tiny elements.

Lastly, after the algorithm completes the claimed subsets, the elements not placed in subsets have total value $\mu(V) - \sum_{i=1}^{r} \mu(\{v_r\}) - W$, which is at least $\mu(V) - \sum_{i=1}^{r} \mu(\{v_r\}) - W^*$, the maximum total value of elements in the subsets of P^* that contain only tiny elements. Our algorithm deals with this value by adding elements to subsets until they become completed, which occurs once their value reaches $\mu(S^*)$. Since each tiny element has value less than $\frac{1}{2}\mu(S^*)$, each completed subset has value less than $\frac{3}{2}\mu(S^*)$, whereas each medium or large subset in P^* containing only tiny elements has value at least $\mu(S^*)$. Thus $C^* < \frac{3}{2}C + \frac{1}{2}$.

If instead $r - i = \lfloor \frac{B^*}{2} \rfloor + 1$, the algorithm attempts to complete an additional subset that already has an intermediate element before creating a new subset with tiny elements, so by a greedy stays ahead argument we still get $B_0^* + B^* + C^* < B_0 + \frac{4}{3}B + 1 + \frac{3}{2}C + \frac{1}{2}$. Combining the (in)equalities derived for $A, A^*, B_0, B_0^*, B, B^*, C, C^*$ yields $\text{OPT} < \frac{3}{2}\text{ALG} + \frac{3}{2} - \frac{1}{2}(l_0 + m_0) - \frac{1}{2}B_0^* - \frac{1}{6}B^*$, which is equivalent to the desired claim since the right-hand side is a multiple of $\frac{1}{2}$. □

Corollary 3.3.5. *The complete graph case of* Rank Percentile Maximization *has a polynomial-time* $\max\left(1, \frac{\frac{3}{2}l' - \frac{1}{2}(l_0+m_0) - \frac{1}{2}B_0^* - \frac{1}{6}B^* + \frac{3}{2}}{\frac{3}{2}l' - \frac{1}{2}(l_0+m_0) - \frac{1}{2}B_0^* - \frac{1}{6}B^* + \frac{5}{2}}\left(1 + \frac{1}{l'}\right)\right)$*-approximation algorithm, where* l_0 *and* m_0 *are the number of vertex singletons that are large and medium, respectively,* B_0^* *and* B^* *are the number of subsets comprised of exactly 2 and 1 intermediate elements in an arbitrary optimal solution, respectively, and* l' *is the number of large subsets, and* l' *is the number of large subsets returned by the algorithm's rank maximization subroutine.*

Proof. We consider 3 cases. If $\mu(V \backslash S^*) < \mu(S^*)$, any partition of V is comprised of small

subsets, the percentile of which is maximized by having just 1 small subset, so the partition $(V \backslash S^*, S^*)$ maximizes the percentile of S^*. Similarly, if $\mu(V \backslash S^*) = \mu(S^*)$, any valid partition of V is comprised of small subsets or one medium subset, the percentile of which is maximized by having just 1 medium subset, so the partition $(V \backslash S^*, S^*)$ maximizes the percentile of S^*. If $\mu(V \backslash S^*) > \mu(S^*)$, there exists a partition with a large subset, and since all partitions are valid, the maximum percentile is achieved by a partition containing l^* large subsets, 0 medium subsets, and 0 small subsets, where l^* is the maximum number of mutually disjoint subsets possible (by the proof of Lemma 3.2.10). In other words, the maximum percentile is $\frac{l^*}{l^*+1}$. Instead, we use the algorithm described for Theorem 3.3.4 to obtain l' large subsets (after applying the preprocessing step in the proof of Lemma 3.2.10 to preclude medium subsets) and combine any remaining elements into one of these subsets, obtaining a percentile of $\frac{l'+1}{l'}$. Hence our approximation ratio in this case is $\frac{l^*}{l^*+1} \cdot \frac{l'+1}{l'}$. We can upper bound the first fraction by adding the same thing to the numerator and denominator, replacing l^* with its upper bound given by the right-hand side of the approximation guarantee in Theorem 3.3.4 plus $\frac{1}{2}$ (since we no longer add 1 to get the rank). This gives the desired factor. □

3.3.2 Linear Component Case

Sometimes, the objects we wish to rank may be organized within each connected component by a single variable, with the similar elements in a connected component being those that have adjacent or consecutive values of the variable. For instance, each connected component may correspond to a television series, with the elements being its episodes chronologically ordered. Subsets in $\mathcal{S}$ may then be interpreted as seasons of a given series, and the above computational problems correspond to optimizing the rank of S^*, (a season of) a fixed series, with respect to hours watched, number of award nominations, etc. While $\mathcal{S}$ still contains exponentially many subsets in general, all four of the above problems are polynomial-time solvable in this special case where all connected components of G are line graphs, called the *linear component* case.

In this case, we will write $V = \{v_1, \ldots, v_n\}$ to mean that each vertex v_i is adjacent to at most v_{i-1} and v_{i+1} (if they exist) and no other vertices, so each connected component of G is a line graph of the form $\{v_j, \ldots, v_{j+k-1}\}$, where k is the number of vertices in the component. Thus S^* constitutes a line subgraph in one of the connected components, though as a preprocessing step for the rank percentile algorithms we discard S^* from the description of V and shift the index of proceeding vertices by $-|S^*|$. This truncates a linear component of G or decomposes it into two linear components, so G is still of the form argued above.

First, the linear component case of Rank Minimization is just a special case of Rank Minimization as defined above, so it is still polynomial-time solvable (see Theorem 3.2.4).

Corollary 3.3.6. *The linear component case of* Rank Minimization *is polynomial-time solvable.*

As for Rank Maximization, this special case reduces to the classical unweighted interval scheduling problem.

Theorem 3.3.7. *The linear component case of* Rank Maximization *is polynomial-time solvable.*

Proof. First suppose G has exactly one connected component. Therefore, G is a line graph with first vertex v_1 and last vertex v_n, and without loss of generality $S^* = \{v_1, \ldots, v_k\}$. We reduce to unweighted interval scheduling as follows. For all $k < i \leq j \leq n$, if $\{v_i, \ldots, v_j\}$ is medium or large, add a job occupying the interval $[i, j + .5]$. Thus the intervals are in one-to-one correspondence with the set of medium or large subsets of V, and two intervals being non-overlapping means that the corresponding subsets of V are disjoint. As a result, maximizing the number of non-overlapping subsets is equivalent to maximizing the number of medium or large subsets with pairwise empty intersection, which is the same as maximizing the number of medium or large subsets in a partition of V that includes S^*. If instead G has multiple connected components, or $G \backslash S^*$ does as a result of S^* being in the middle of a connected component, perform this reduction for each connected component and take the union of the optimal set of jobs over each instance; this still maximizes the number of medium or large subsets with pairwise empty intersection because it does so for each connected component, and a subset in $\mathcal{S}$ may not intersect multiple components. The preprocessing step has runtime $O(n^2)$, and the greedy subroutine for unweighted interval scheduling is then $O(n^2 \log n)$, so our algorithm is $O(n^2 \log n)$. □

For the percentile problems of this special case, we must also take care to the fact that when modifying a partition of V, the percentile of S^* experiences "inertia" based on how many subsets are already in the partition. Due to the one-dimensional structure of subsets in this case, however, we may optimize the percentile with a relatively efficient dynamic programming approach.

Theorem 3.3.8. *The linear component case of* Rank Percentile Minimization *and* Rank Percentile Maximization *are polynomial-time solvable.*

Proof. Consider the following algorithm for Rank Percentile Minimization under this case.

Algorithm 2 Given an instance (V, E, μ, S^*) of Rank Percentile Minimization in the linear component case, compute the minimum percentile of S^*

---

for $1 \leq i \leq n$ **do**
  for $i < j \leq n$ **do**

$$C[i,j] = \begin{cases} \frac{1.5}{2} & \text{if } \{v_i, \ldots, v_j\} \text{ is large} \\ \frac{1}{2} & \text{if } \{v_i, \ldots, v_j\} \text{ is medium} \\ \frac{.5}{2} & \text{if } \{v_i, \ldots, v_j\} \text{ is small} \\ \infty & \text{otherwise} \end{cases}$$

  end for
  $\mathrm{OPT}[i, 1] = C[1, i]$
end for
for $2 \leq k \leq n$ **do**
  for $1 \leq j \leq n$ **do**
    $\mathrm{OPT}[j, k] = \frac{1}{k+1}(\inf_{1 \leq i < j}\{k\mathrm{OPT}[i, k-1] + 2C[i+1, j] - .5\})$
  end for
end for
return $\min_{1 \leq k \leq n}\{\mathrm{OPT}[n, k]\}$

To see that this algorithm is correct, we argue that $\mathrm{OPT}[j, k]$ gives the minimum percentile of S^* under a valid partition with exactly k subsets other than S^* in the problem instance where the vertex set is $\{v_1, \ldots, v_j\} \cup S^*$ and the similarity relation is $E|_{(\{v_1,\ldots,v_j\}\cup S^*)\times(\{v_1,\ldots,v_j\}\cup S^*)}$. (We also adopt the convention that if there is no such partition, the minimum percentile is ∞.) First, when $c = 1$, notice that we have $\mathrm{OPT}[j, 1] = C[1, j]$, which by definition gives the percentile of S^* under the only partition of $\{v_1, \ldots, v_j\} \cup S^*$ with 1 subset other than S^*: the trivial partition $(\{v_1, \ldots, v_j\}, S^*)$. (If $\{v_1, \ldots, v_j\} \notin \mathcal{S}$, then there is no such valid partition of $\{v_1, \ldots, v_j\} \cup S^*$, and we set $C[1, j] = \infty$.) Suppose the claim holds for $k = k' - 1 \in \mathbb{N}$ and all $1 \leq i < j$. Observe that if $k' \geq j$, then there any expression of the form $\mathrm{OPT}[i, k' - 1]$ will be ∞ by hypothesis since we may not partition $i < j$ elements into k' non-empty subsets; correspondingly, the infimum used to define $\mathrm{OPT}[j, k']$ is infinite. Otherwise, $k' - 1 \leq j$.

Because G has linear components, the valid partitions of $\{v_1, \ldots, v_j\} \cup S^*$ with k' subsets other than S^* are exactly the valid partitions of $\{v_1, \ldots, v_i\} \cup S^*$ plus the subset $[i + 1, j]$,

over all $1 \leq i < j$ such that $\{v_{i+1}, \ldots, v_j\} \in \mathcal{S}$. We claim that under the latter, S^* has minimum percentile equal to $\frac{1}{k'+1}(k'\text{OPT}[i, k'-1]+2C[i+1, j]-.5)$. Indeed, $k'\text{OPT}[i, k'-1]$ gives the number of large subsets plus .5 times the number of medium subsets in a given valid partition of $\{v_1, \ldots, v_i\} \cup S^*$ plus .5; $2C[i+1, j]$, $[i+1, j]$. Now subtracting .5 and then multiplying by $\frac{1}{k'+1}$, we obtain the expression for the percentile of S^* under a valid partition of $\{v_1, \ldots, v_j\} \cup S^*$ with k' subsets. Similarly, for given i as above, the percentile under a partition with k' subsets other than S^* and the last subset $\{v_{i+1}, \ldots, v_j\}$ is a decreasing function of the percentile under the corresponding partition of $\{v_1, \ldots, v_i\} \cup S^*$ with $k'-1$ subsets other than S^*, so the former is minimized by such a corresponding partition achieving minimum percentile, i.e., $\text{OPT}[i, k'-1]$. Taking the infimum over all $1 \leq i < j$ results in minimizing the percentile under a partition of $\{v_1, \ldots, v_j\} \cup S^*$ with k' subsets other than S^*. (If the infimum is ∞, there is no valid partition of $\{v_1, \ldots, v_j\} \cup S^*$ with k' subsets other than S^*).

Finally, we note that there may be at most n subsets in a partition of $V = \{v_1, \ldots, v_n\}$, so $\min_{1 \leq k \leq n}\{\text{OPT}[n, k]\}$ is the minimum percentile of S^* under a valid partition of V.

To solve Rank Percentile Maximization, we simply replace ∞ with $-\infty$, inf with sup, and min with max in Algorithm 2; the proof of correctness is analogous. □

3.3.3 Uniform Value Case

Another natural special case of the problem is where μ assigns the same value to each element of V; concretely, for all $v \in V$, $\mu(\{v\}) = 1$ (without loss of generality). Let $|S^*| = k \in \mathbb{N}$. Then, a subset S is large if $|S| > |S^*|$, medium if $|S| = |S^*|$, and small if $|S| < |S^*|$. We call this the *uniform value* case.

To see an application of this, think of each element of V as the physical location of a specific type of resource, like a hospital or a polling place, and E the set of roads connecting them. Now each $S \in \mathcal{S}$ can be thought of as a region bounded by roads, and the rank or percentile of S^* under a partition indicates how well-resourced S^* is as a region.

By Theorem 3.2.4, the uniform value case of Rank Minimization is polynomial-time solvable, and because each element of V has the same value, all singletons are small subsets, so the partition that minimizes rank also maximizes the number of subsets in a valid partition of V while having no medium or large subsets, hence achieving the minimum

percentile of S^* as well. Therefore, Rank Percentile Minimization is polynomial-time solvable as well.

Given a graph G_0 with kt vertices for any $t \in \mathbb{N}$, if we let G consist of G_0 with k additional vertices forming S^*, then finding a rank of at least $t+1$ is the same as the NP-complete problem $Vk(\text{connected})$, so the uniform value case of Rank Maximization is NP-hard [23, 17].

We can also note that the uniform value case of Rank Maximization is the same as the special case of the optimization version of k-Set Packing where the set-system $(V, \mathcal{S})$ consists of the vertices of an undirected graph G with its subsets of size k that form a connected subgraph of G. In other words, given an undirected graph G, our objective is to exhibit a maximum-cardinality union of vertex-disjoint induced subgraphs of G with k vertices (henceforth referred to as *k-components*). Moreover, we have a $\frac{k+2}{3}$-approximation algorithm for k-Set Packing [36].

If G is a connected circulant graph, then it is known that G contains a Hamiltonian path [13] and hence contains $\lfloor \frac{n}{k} \rfloor$ disjoint k-components, since we can place the first k vertices in a k-component and recurse. We give an algorithmic proof of this fact, allowing us to compute a rank-maximizing partition of V in polynomial-time. Denote $G = C_n(s_1, \ldots, s_l)$, where $s_1, \ldots, s_l < \frac{n}{2}$ are called *jumps*; writing $V = \{0, 1, \ldots, n-1\}$, we have $(i,j) \in E \iff j - i \equiv \pm s_r \pmod{n}$.

Theorem 3.3.9. *The uniform value case of* Rank Maximization *with the additional restriction that $G \backslash S^*$ be connected and circulant is polynomial-time solvable.*

Proof. We prove the stronger statement that there exists a Hamiltonian path in $G \backslash S^*$ by strong induction on the number of jumps l. If $l = 1$, then $G \backslash S^*$ is connected if and only if $\gcd(n, s_1) = 1$, but this means that $\{s_1\}$ generates the additive group $\mathbb{Z}/n\mathbb{Z}$, and so the path $(0, s_1, 2s_1, \ldots, (n-1)s_1)$ is Hamiltonian. Now for fixed $l > 1$, assume the statement holds for $l-1, l-2, \ldots, 1$. Consider the following circulant subgraph of $G \backslash S^*$, $C_n(s_1)$. It consists of $g_1 = \frac{\text{lcm}(n, s_1)}{s_1} = \gcd(n, s_1)$ cycles (the number of s_1 increments needed to return to the initial vertex), each having $b_1 = \frac{n}{g_1}$ vertices by symmetry. Also note that the vertices $0, 1, \ldots, g_1 - 1$ are in g_1 distinct cycles (because n and s_1 are both multiples of g_1 and hence $ts_1 - n$ is divisible by g_1 for any $t \in \mathbb{Z}$, so no two elements of a cycle may have a difference of greater than $-g_1$ but less than $g_1 \pmod{n}$). Thus label the cycles as $0, 1, \ldots, g_1 - 1$ corresponding to which of these vertices it contains. Now

in each cycle, perform edge contractions until each cycle only has one vertex, denoted $0, 1, \ldots, g_1 - 1$ as above. Note that the vertices in cycle c are described by the set $\{c + ts_1 \pmod{n}: t \in \mathbb{Z}\}$, and so the set of possible differences between two vertices in cycles c_1 and $c_2 \pmod{n}$ is given by $\{c_2 - c_1 + ts_1 \pmod{n}: t \in \mathbb{Z}\} = -\{c_1 - c_2 + ts_1 \pmod{n}: t \in \mathbb{Z}\}$.

Fix $x \in \{2, \ldots, l\}$ and consider s_x. Note that given two cycles $c_1, c_2 \in \{0, 1, \ldots, g_1 - 1\}$, $c_2 - c_1 \in \{-\lfloor \frac{g_1}{2} \rfloor, -\lfloor \frac{g_1}{2} \rfloor + 1, \ldots, 0, 1, \ldots, \lfloor \frac{g_1}{2} \rfloor\} \pmod{n}$. Now, if there exists $t_0 \in \mathbb{Z}$ such that $c_2 - c_1 + t_0 s_1 = \pm s_x \pmod{n}$, then $-(c_2 - c_1) + -t_0 s_1 = \mp s_x \pmod{n}$. Moreover, replacing $\pm t_0 s_1$ with another integer $\pmod{n}$ results in adding a non-zero multiple of s_1, hence a non-zero multiple of g_1, to the left-hand side of the equation $\pmod{n}$, but $\pm(c_2 - c_1)$ is uniquely determined by its remainder mod n. Therefore, a jump of s_x either does not induce any edges in the contracted graph or induces an edge between c_2 and c_1 in the contracted graph exactly when $c_2 - c_1 = \pm s'_x$ for unique $s'_x \in \{-\lfloor \frac{g_1}{2} \rfloor, -\lfloor \frac{g_1}{2} \rfloor + 1, \ldots, 0, 1, \ldots, \lfloor \frac{g_1}{2} \rfloor\}$.

Hence the contracted graph is circulant (and clearly it is connected) with at most $l - 1$ distinct jumps, so by the inductive hypothesis there exists a Hamiltonian path in that graph. Moreover, the above expressions show that, if two vertices from different cycles are adjacent in $G \backslash S^*$, then by adding the multiples $s_1, 2s_1, \ldots, (b_1 - 1)s_1 \pmod{n}$ to both of the vertices, we see that every vertex in one cycle has an edge to a vertex in the other cycle in $G \backslash S^*$. As a result, by replacing each vertex in the Hamiltonian path of the contracted graph with a Hamiltonian path in the corresponding cycle, we obtain a Hamiltonian path in $G \backslash S^*$ (since in $G \backslash S^*$, we can move between cycles that share an edge by taking an edge from any vertex in the first cycle). □

Furthermore, in the setting of the preceding theorem, unless there are no large subsets in G (in which case the problem is trivial), Lemma 3.2.10 tells us to just focus on maximizing the number of large subsets to maximize the percentile.

Corollary 3.3.10. *The uniform value case of* Rank Percentile Maximization *with the additional restriction that $G \backslash S^*$ be connected and circulant is polynomial-time solvable.*

3.4 Variants of the Problem

3.4.1 Equivalence Class Variant

For some of the examples given above, it makes more sense to study the following *equivalence class* variant of the above problems obtained by requiring that E be an equivalence relation

and redefining $\mathcal{S}$ as the set of singletons and equivalence classes of V. It turns out that, while there are still exponentially many possible partitions, this additional structure results in all four of the above problems becoming polynomial-time solvable.

Corollary 3.4.1. *The equivalence class variant of* Rank Minimization *is polynomial-time solvable.*

Proof. Without the additional restrictions on $\mathcal{S}$, the algorithm in Theorem 3.2.4 would just take as the partition sets the equivalence classes associated with each large element along with any remaining singletons. The minimum rank may only increase by removing subsets from $\mathcal{S}$, so this partition still achieves the minimum rank. □

Theorem 3.4.2. *The equivalence class variant of* Rank Maximization *is polynomial-time solvable.*

Proof. Consider the partition of V obtained by taking each equivalence class exactly when at most one of its elements are large—excluding the equivalence class containing S^* (for which the partition is predetermined). Then reverting any of these selected equivalence classes into singletons does not increase the rank of S^* (since the value of a subset containing a large element does not increase), whereas consolidating singletons sets in a non-selected equivalence class decreases the rank (since by construction there are at least two large elements), and these are the only admissible deviations from this partition. □

As for optimizing the percentile of S^*, respecting equivalence classes greatly simplifies the space of admissible partitions, decomposing it into linearly many choices: one for each equivalence class.

Theorem 3.4.3. *The equivalence class variants of* Rank Percentile Minimization *and* Rank Percentile Maximization *are polynomial-time solvable.*

Proof. We argue for Rank Percentile Minimization; Rank Percentile Maximization follows by a symmetric argument. Consider the following algorithm: initialize $P = P_0$ to be the partition consisting of singletons and S^*, and $p = p_0$ the percentile under P. While there exists an equivalence class not in P for which $\frac{l+.5m-1}{l+m+s-1} > p$, add such an equivalence class to P by combining the corresponding singletons and update p to the percentile under P.

Upon termination, we claim that p is the minimum percentile p^* of S^* under a valid partition, and that it is achieved by P. To see this, note that a partition with percentile p^* must contain any equivalence class for which $\frac{l+.5m-1}{l+m+s-1} > p^*$; otherwise, Lemma 3.2.3 implies that combining the equivalence class's singletons decreases the percentile, contradicting optimality of p^*. Any

percentile p realized during the execution of the algorithm necessarily satisfies $p \geq p^*$, so the equivalence classes it adds are part of an optimal partition P^*. To see that the rest of the equivalence classes are not part of such a partition, observe that after the algorithm's termination we have that for any equivalence class not in P, $\frac{l+.5m-1}{l+m+s-1} \leq p$; if the inequality is strict, Lemma 3.2.3 states that adding the corresponding subset to P would increase the percentile, and then the criterion for such only becomes weaker if such subsets have already been added by repeated application of the lemma. If it is equality, Lemma 3.2.2 reveals adding the subset does not change the percentile and hence may be included or excluded. Thus P consists of exactly the subsets in an optimal partition P^*, so the algorithm is correct. For runtime, we can sort, compare, and compute percentiles in polynomial time (e.g., by cross multiplication). □

3.4.2 Hierarchical Category Variant

It is natural to extend the formalism in the preceding section to allow for subcategories of arbitrary depth in classifying elements and to allow an element to be classified under multiple (sub)categories, each instance having a different weight (value under μ). In order to do this, we modify the set-up of our rank optimization problems as follows:

- G is comprised of the leaves of a rooted forest consisting of trees $T_1, \ldots, T_m$ in which no nodes have exactly 1 child.
- The root of each tree T_j represents a *category* of V identified with all of its descendants that are leaves. Each descendant of the root that in turn has descendants is a *subcategory* (of each of its ancestors) identified with all of its descendants that are leaves (we also still consider it a category of V). (In the case where the root is a leaf, we no longer call it a category of V, but it is still an element of V).
- The elements of V are partitioned into classes $\tilde{V}$, each one representing all instances of a particular item under the category given by its parent node.
- The rank and percentile of S^* are only optimized over valid partitions of V such that each subset (including S^*) is a category of V or the elements in a class in $\tilde{V}$ that are not in a category subset in the partition.

Sometimes, we also add another node v_0 whose children are the roots of the above trees, yielding a composite tree T'.

These rules are motivated by several real-world examples of ranking.

First, food manufacturers have numerous degrees of freedom when listing ingredients on packaged food labels. In the United States, for instance, even though the Food and Drug Administration (FDA) requires that all ingredients be listed in descending weight order, an ingredient consisting of multiple subingredients may either be listed on the label according to its total weight (with the subingredients listed in descending weight order using parentheses) or its subingredients may be listed independently without reference to the original ingredient (including recursive application of parentheses), cf. *21 CFR 101.4(b)(2)* [39]. One consequence of these rules is that an ingredient with no subingredients may be listed in multiple places on a food label according to its weight deriving from that specific place. For example, "sugar" may be listed as a subingredient of "milk chocolate" (its order among the subingredients determined by its weight in the milk chocolate) and as its own ingredient in the ingredient list (its order determined by its weight that is not part of the milk chocolate).

Similarly, there are many ways to organize the myriad greenhouse emissions that contribute to climate change, and the same type of emission can be broken up into the different uses that result in it being emitted. For this, the EPA uses the "Common Reporting Tables" (CRT) prescribed by the Paris Agreement. As an example, "Incineration of Waste" is part of the broader CRT Source Category 1A consisting of energy industry sources, but conversely can be decomposed further based on the gas emitted, e.g., CO_2 (even if these gases are released in other contexts) [2].

As for the computational nature of this setting, this is the first setting where just minimizing rank is NP-hard, resulting in all four of our problems being intractable and introducing the need for additional structure.

Theorem 3.4.4. *The hierarchical category variant of* Rank Minimization *is* NP*-hard.*

Proof. We reduce from X3C. Given an instance with n elements and m subsets, we create a tree T_m of depth 1 for each subset, where there is one leaf node for each element in the corresponding subset. $\tilde{V}$ is defined by declaring all leaf nodes for a given element equivalent (across all the trees we create). Lastly, create a tree with one node v^*, which will be the only element in S^*. Set $\mu(S^*) = 5$ and $\mu(\{v\}) = 2, v \neq v^*$. Therefore, each category (subset in the X3C instance) and each class in $\tilde{V}$ have value 6 under μ, so they are large subsets, but a subset of a class with just 2 of its representatives has value 4 under μ, which is a small subset. We claim there exists an exact cover if and only if the minimum rank of S^*

is $\frac{n}{3}+1$. For the forward direction, let each category pertaining to a subset in the exact cover be in the partition, with no other categories in the partition. The cover has size $\frac{n}{3}$, and the remaining elements form subsets of classes in $\tilde{V}$ of size 2, hence not large. Thus this partition achieves rank $\frac{n}{3}+1$. For the reverse direction, we note that there cannot be more than $\frac{n}{3}$ categories selected in the partition achieving minimum rank, but if there are k fewer categories selected, the partition has at least $3k-k$ more large subsets than $\frac{n}{3}$, so we must have $\frac{n}{3}$ category subsets. But now, if they do not form an exact cover, then the leftover part of the class for an element not covered is a large subset in the partition, a contradiction. □

Theorem 3.4.5. *The hierarchical category variant of* Rank Maximization *is* NP*-hard.*

Proof. We reduce from X3C. Given an instance with n elements and m subsets, we create a tree T_m of depth 1 for each subset, where there is one leaf node for each element in the corresponding subset. $\tilde{V}$ is defined by declaring all leaf nodes for a given element equivalent (across all the trees we create). Lastly, create a tree with one node v^*, which will be the only element in S^*. Set $\mu(S^*)=3$ and $\mu(\{v\})=2, v\neq v^*$. Therefore, each category (subset in the X3C instance) and each class in $\tilde{V}$ have value 6 under μ, so they are large subsets, but a subset of a class with just 1 of its representatives has value 2 under μ, which is a small subset. We claim there exists an exact cover if and only if the maximum rank of S^* is $\frac{4n}{3}+1$. For the forward direction, we can choose each subset in the X3C instance that is part of the exact cover, which keeps each element's class having 2 elements, resulting in $\frac{n}{3}+n$ large subsets in total. For the reverse direction, note that at least $\frac{n}{3}$ category subsets must be in the partition, but that if k more are selected, then at least $3k$ elements are covered twice, so $\frac{4n}{3}$ large subsets is not possible. Thus the number is exactly $\frac{n}{3}$, and they must only cover each element once so that the corresponding classes have 2 representatives left and are therefore large. Hence these subsets form an exact cover. □

Moreover, by observing that the number of small and large subsets is optimized if and only if an exact cover exists, these reductions establish that the hierarchical category variants of Rank Percentile Minimization and Rank Percentile Maximization are NP-hard.

3.4.2.1 Bounded Degree and $|\tilde{V}|=|V|$ Case

The hardness results in this subsection arose due to the combinatorial complexity introduced by allowing representatives to be part of multiple subingredients with different siblings. We now show that if this is disallowed, i.e. $|\tilde{V}|=|V|$ (meaning there is only one instance of each item), then the rank problems become easy and the percentile ones are polynomial-time solvable if the maximum degree of G is bounded by a constant.

First, for rank minimization, the procedure is the following modified breadth-first search procedure on T'. Add the root to a list called E, set rank equal to 1, and initialize a partition $P = S^*$. While there is a node in E, pop such a node v from E and set a variable called D equal to false. If v is not equal to S^* and there is not a path from v to S^* that only goes away from the root (i.e. that can only move from a node to one of its child nodes), then set D equal to true. If D is true and the value of at least one leaf node that is a descendant of v is at least the value of S^* under μ, update $P = P \cup v$ and increment rank. Else if D is true, put all leaf nodes that are descendants of v into P as singletons. Else if D is false, add all child nodes of v to E. To see correctness, note that for each node v encountered in E such that D is true, if one of its leaf nodes is medium or large, we add a subset consisting of all the leaf nodes in the subtree with v as the root, and otherwise we leave each leaf node in the subtree as a singleton in the partition. If the former condition holds, then at least one subset from the subtree will be medium or large, and we ensure this lower bound is achieved; if the latter condition holds, none of the subsets from the subtree will be medium or large.

For rank maximization, the procedure is as follows. For each category that does not intersect S^*, exclude it from the final partition if and only if at least one of its children is medium or large. The resultant partition is well-defined because leaf nodes can always form singleton subsets in the partition, and a category that is included in the partition must be medium or large with no such children or small itself, so none of its descendants may be medium or large (due to additivity of μ) and hence all will be excluded from the partition except for leaf nodes. Similarly, correctness follows from a simple inductive argument.

For rank percentile minimization and maximization, a challenge is caused by the fact that we do not know the optimal percentile (and hence whether excluding a category improves the percentile) a priori, and even if we did, the number of subsets we have added to any partition built iteratively will affect the magnitude of changes from subsequent subsets to the percentile. Therefore, there are several senses in which locally optimal decisions do not lead to a globally optimal solution. However, it is feasible to find the globally optimal solution if the degree of each category vertex is not too large; the hierarchical category variant of Rank Percentile Minimization and Rank Percentile Maximization are fixed-parameter tractable.

Theorem 3.4.6. *The hierarchical category variant of* Rank Percentile Minimization *and* Rank Percentile Maximization *under the restriction that* $|\tilde{V}| = |V|$ *are solvable*

in time $O\left(\left(\frac{e \cdot n}{\min\{d,m\}}\right)^{\max\{d,m\}}\right)$*, where* d *is the maximum tree degree of a vertex in* G*.*

Proof. Let $\mathrm{OPT}[u, c]$ be the minimum percentile when partitioning the elements of category u into c subsets (if no such partition exists, it is instead equal to ∞). Suppose u has subcategories $u_1, \ldots, u_r$ as children (excluding S^* as a subcategory) and $c > 1$. Then the following recurrence holds:

$$\mathrm{OPT}[u, c] = \min_{\substack{c_1 \geq 1, \ldots, c_r \geq 1 \\ c_1 + \cdots + c_r = c}} \left\{ \frac{c_1 + 1}{c+1} \mathrm{OPT}[u_1, c_1] + \cdots + \frac{c_r + 1}{c+1} \mathrm{OPT}[u_r, c_r] - \frac{r-1}{2(c+1)} \right\}$$

To see this, note that there are two options for category u in a partition of V: it may be one subset in the partition (corresponding to the below case), or each subcategory of u may be one subset in the partition (or subdivided into multiple subsets according to the above rules). Therefore, we minimize over all partitions considered in the hierarchical category variant. The coefficients in the recurrence are selected to give the percentile of the partition obtained by combining the choice of subsets for each subcategory of u. If u is a leaf node or $c = 1$, we instead have

$$\mathrm{OPT}[u, c] = \begin{cases} \frac{1.5}{2} & \text{if } u \text{ is large}, u \notin S^*, \text{and } c = 1 \\ \frac{1}{2} & \text{if } u \text{ is medium}, u \notin S^*, \text{and } c = 1 \\ \frac{.5}{2} & \text{if } u \text{ is small}, u \notin S^*, \text{and } c = 1 \\ \infty & \text{otherwise} \end{cases}.$$

As for computation, each choice of $(c_1, \ldots, c_r)$ in the second min operator requires $O(1)$ runtime, and there are $\binom{c-1}{r-1}$ such choices by a stars-and-bars argument. Repeating for at most n categories and using that $\binom{c-1}{r-1} \leq (\frac{e \cdot (c-1))}{r-1})^{r-1}$ with $c \leq n$ and $r \leq \max\{d, m\}$, we get the claimed runtime (since $\max\{d, m\} \leq \frac{en}{e} = n$). We return

$$\min_{1 \leq c \leq n} \mathrm{OPT}[v_0, c].$$

To solve the hierarchical category variant of Rank Percentile Maximization, replace ∞ with $-\infty$ and min with max in the above algorithm; the proof of correctness is analogous. □

3.4.3 Grid Variants

An interesting modification of the preceding, where G has no natural embedding in $\mathbb{R}^n$, is to view the graph G as a two-dimensional grid, perhaps representing locations or equally spaced points on a map or a discretization thereof. We also allow a set of vacancies V_- in

the grid, representing points that would not be actual locations within G (however interpreted). Therefore, without loss of generality we may regard V as $[1,l] \times [1,w] \backslash V_-$, where $V_- \subset [1,l] \times [1,w]$ and $n = |V| = lw - |V_-|$. Lastly, two points $(a,c),(b,d) \in V$ are similar if they occupy adjacent positions in the grid: $((a,c),(b,d)) \in E \iff |b-a|+|d-c| = 1$.

For instance, if G represents the United States as projected on a map, then we may leave some vacancies for Canada and the Atlantic Ocean instead of, say, stacking New England on top of New York, Pennsylvania, and Ohio.

Denote $\square_{abcd} := [a,b] \times [c,d]$, the rectangular grid spanning x-coordinates a to b and y-coordinates c to d (inclusive).

In the below variants of the problem, we study partitions of the grid $[1,l] \times [1,w]$ into axis-parallel rectangles. By taking the intersection of each rectangle with V, we obtain an induced partition of V. As usual, we seek to optimize the percentile of S^* (a subgrid of V that without loss of generality does not contain any vacancies) under a valid partition of V; we just place restrictions on which valid partitions may be considered, as described for each of the proceeding variants below.

For both variants, we specify a collection of subsets of V_- representing valid subsets of vacancies to place in a rectangle with elements of V. This may be specified in polynomial space by giving a list of rectangles $\mathcal{R} \subset \mathcal{P}(\{\square_{abcd} : 1 \le a \le b \le l, 1 \le c \le d \le w\})$ that may be included in the partition of V, with the caveat that if two rectangles contain elements of V and the same subset of V_-, they are either both allowed or both disallowed (and rectangles comprised only of elements of V are allowed). In other words, $\square_{abcd} \cap V_- = \emptyset \implies \square_{abcd} \in \mathcal{R}$ and $\square_{abcd} \cap V_- = \square_{a'b'c'd'} \cap V_- \implies (\square_{abcd} \in \mathcal{R} \iff \square_{a'b'c'd'} \in \mathcal{R})$.

3.4.3.1 Grid with Hierarchical Rectangles Variant

In the *grid with hierarchical rectangles* variant, we consider the restriction where the only admissible partitions of V are those induced by hierarchically partitioning the grid into rectangles with axis-parallel edges. That is, in each step, we take a subgrid from a previous step ($[1,l] \times [1,w]$ for the first step) and draw a horizontal or vertical line between two rows or columns thereof, respectively, resulting in the cut subgrid becoming two subgrids separated by the line instead. Upon termination, each subgrid containing elements of V must be in $\mathcal{R}$, which induces a partition of V.

Theorem 3.4.7. *The grid with hierarchical rectangles variant of* Rank Minimization *and* Rank Maximization *are polynomial-time solvable.*

Proof. Consider the following algorithm.

Algorithm 3 Given an instance (V, E, μ, S^*) of Rank Minimization in the grid with hierarchical rectangles variant, compute the minimum rank of S^*

Initialize four-dimensional table OPT of size $l \times l \times w \times w$ with NULL entries

Compute-Opt$(1, l, 1, w)$

procedure Compute-Opt(a, b, c, d)

  if OPT$[a, b, c, d]$ is not NULL **then**

    return OPT$[a, b, c, d]$

  else if $\square_{abcd} = S^*$ or $\square_{abcd} \subset V_-$ **then**

    OPT$[a, b, c, d] = 0$

  else if $(\square_{abcd} \cap S^* \neq \emptyset$ and $\square^{\mathrm{c}}_{abcd} \cap S^* \neq \emptyset)$ **then**

    OPT$[a, b, c, d] = \infty$

  end if

  $$c_b \leftarrow \begin{cases} 1 & \text{if } \square_{abcd} \text{ is large, not } S^* \subset \square_{abcd}, \text{ and } \square_{abcd} \in \mathcal{R} \\ 0 & \text{if } \square_{abcd} \text{ is medium or small, not } S^* \subset \square_{abcd}, \text{ and } \square_{abcd} \in \mathcal{R} \\ \infty & \text{otherwise} \end{cases}$$

  $c_v \leftarrow \infty$

  $c_h \leftarrow \infty$

  if $a < b$ **then**

    $c_v \leftarrow \min_{a \le i < b} \{$Compute-Opt$(a, i, c, d)$ + Compute-Opt$(i + 1, b, c, d)\}$

  end if

  if $c < d$ **then**

    $c_h \leftarrow \min_{c \le j < d} \{$Compute-Opt$(a, b, c, j)$ + Compute-Opt$(a, b, j + 1, d)\}$

  end if

  OPT$[a, b, c, d] = \min\{c_b, c_v, c_h\}$

  return OPT$[a, b, c, d]$

end procedure

We claim that for all $1 \le a \le b \le l, 1 \le c \le d \le w$, Compute-Opt$(a, b, c, d)$ gives the minimum number of large subsets of $\square_{abcd} \backslash V_-$ when $\square_{abcd} \backslash V_-$ is hierarchically partitioned into subsets in $\mathcal{R}$ with the further restriction that S^* is one of the subsets (or that S^* does not intersect $\square_{abcd}$ and its complement, in which case Compute-Opt(a, b, c, d) should

be infinity). Taking $a = 1, b = l, c = 1, d = w$ will then prove the correctness of the algorithm.

First, if $\square_{abcd} = S^*$ or $\square_{abcd} \subset V_-$, then the subgrid we are partitioning is vacuous and hence there are 0 large subsets. Similarly, if S^* intersects $\square_{abcd} \backslash V_-$ but also its complement, S^* may not be one of the subsets in the former and the algorithm must assign a value of infinity to COMPUTE-OPT(a, b, c, d).

Otherwise, $\square_{abcd}$ does not intersect S^* or it properly contains S^*, and in both cases it includes elements of V that need to be partitioned. There are two options. One, we may leave $\square_{abcd}$ as one subset, but only if it is in $\mathcal{R}$ and does not contain S^*. In that case, the number of large subsets is 1 if $\square_{abcd}$ is large and 0 otherwise (if it is not possible, $\square_{abcd}$ is not a singleton and hence the second option will be possible). Two, we may subdivide $\square_{abcd}$ by cutting it vertically (assuming $a \neq b$) in one of $b - a$ positions or horizontally (assuming $c \neq d$) in one of $d - c$ positions and only further partitioning each of these subdivisions (no subsets crossing this cut). In that case, the minimum number of large subsets is the minimum sum (over all positions to make the cut) of the minimum number of large subsets for each resultant subgrid, since a hierarchical partition of $\square_{abcd}$ with a horizontal or vertical cut across which there may be no subsets is exactly comprised of a hierarchical partition of each subgrid. Finally, the minimum number of large subsets of $\square_{abcd} \backslash V_-$ is the minimum number resulting from each option. Hence if we suppose that COMPUTE-OPT(a', b', c', d') has the desired behavior for all $\square_{a'b'c'd'} \subsetneq \square_{abcd}$, the claim follows by induction on the size of the subgrid $(b - a)(d - c)$ (if it reaches 1, then no recursive calls are made).

To solve the grid with hierarchical rectangles variant of RANK MAXIMIZATION, we simply replace ∞ with $-\infty$, min with max, "large" with "large or medium", and "medium or small" with "small" in Algorithm 3; the proof of correctness is analogous. □

Theorem 3.4.8. *The grid with hierarchical rectangles variant of* RANK PERCENTILE MINIMIZATION *and* RANK PERCENTILE MAXIMIZATION *are polynomial-time solvable.*

Proof. Consider the following algorithm.

Algorithm 4 Given an instance (V, E, μ, S^*) of Rank Percentile Minimization in the grid with hierarchical rectangles variant, compute the minimum percentile of S^*

Initialize five-dimensional table OPT of size $l \times l \times w \times w \times n$ with NULL entries
return $\min_{1 \le k^* \le n} \{\text{Compute-Opt}(1, m, 1, n, k^*)\}$

procedure Compute-Opt(a, b, c, d, k)
  if OPT$[a, b, c, d, k]$ is not NULL **then**
    return OPT$[a, b, c, d, k]$
  else if $(\square_{abcd} = S^* \text{ or } \square_{abcd} \subset V_-)$ and $k = 0$ **then**
    OPT$[a, b, c, d, k] = 0$
  else if $(\square_{abcd} \cap S^* \neq \emptyset \text{ and } \square^{\text{c}}_{abcd} \cap S^* \neq \emptyset)$ or $k = 0$ **then**
    OPT$[a, b, c, d, k] = \infty$
  end if
  $c_b \leftarrow \begin{cases} \frac{1.5}{2} & \text{if } \square_{abcd} \text{ is large, not } S^* \subsetneq \square_{abcd}, \square_{abcd} \in \mathcal{R}, \text{ and } k = 1 \\ \frac{1}{2} & \text{if } \square_{abcd} \text{ is medium, not } S^* \subsetneq \square_{abcd}, \square_{abcd} \in \mathcal{R}, \text{ and } k = 1 \\ \frac{.5}{2} & \text{if } \square_{abcd} \text{ is small, not } S^* \subsetneq \square_{abcd}, \square_{abcd} \in \mathcal{R}, \text{ and } k = 1 \\ \infty & \text{otherwise} \end{cases}$
  $c_v \leftarrow \infty$
  $c_h \leftarrow \infty$
  if $a < b$ **then**
    $c_v \leftarrow \frac{1}{k+1} \min_{a \le i < b, 0 \le k_h \le k} \{(k_h + 1)\text{Compute-Opt}(a, i, c, d, k_h) + (k - k_h + 1)\text{Compute-Opt}(i + 1, b, c, d, k - k_h) - \frac{1}{2}\}$
  end if
  if $c < d$ **then**
    $c_h \leftarrow \frac{1}{k+1} \min_{c \le j < d, 0 \le k_v \le k} \{(k_v + 1)\text{Compute-Opt}(a, b, c, j, k_v) + (k - k_v + 1)\text{Compute-Opt}(a, b, j + 1, d, k - k_v) - \frac{1}{2}\}$
  end if
  OPT$[a, b, c, d, k] = \min\{c_b, c_v, c_h\}$
  return OPT$[a, b, c, d, k]$
end procedure

The proof of correctness is similar to that for Algorithm 3, except that we are now minimizing percentile instead of rank. Accordingly, we claim that Compute-Opt(a, b, c, d, k) gives the minimum percentile of S^* when $\square_{abcd} \backslash V_-$ is hierarchically partitioned into exactly $k \ge 0$ subsets in $\mathcal{R}$ other than S^* with the further restriction that S^* be one of the subsets (or that S^* not intersect $\square_{abcd}$ and its complement, in which case Compute-Opt(a, b, c, d, k)

should be infinity). Moreover, if there is no such partition with our value of k, COMPUTE-OPT(a, b, c, d, k) should be infinity. We also adopt the convention that if $k = 0$ and there are no non-S^* elements of V in $\square_{abcd}$, COMPUTE-OPT(a, b, c, d, k) should be infinity. We call this the *minimum percentile of S^* for $\square_{abcd}$*. Taking $a = 1, b = l, c = 1, d = w$ for each $1 \leq k^* \leq n$ will then prove the correctness of the algorithm, since the minimum percentile of S^* in the original instance is achieved by the minimum percentile of S^* under a valid partition of V with the number of subsets other than S^* being between 1 and n, so the algorithm returns the minimum percentile of S^* over all valid partitions of V.

First, if $\square_{abcd} = S^*$ or $\square_{abcd} \subset V_-$, then the subgrid we are partitioning is vacuous and we must have $k = 0$ for such a partition. Similarly, if S^* intersects $\square_{abcd} \backslash V_-$ but also its complement, S^* may not be one of the subsets in the former and the algorithm must assign a value of infinity to COMPUTE-OPT(a, b, c, d, k). Lastly, if $k = 0$ but there are non-S^* elements of V in $\square_{abcd}$, then we cannot put those element in any partition subset and the algorithm must assign a value of infinity to COMPUTE-OPT(a, b, c, d, k).

Otherwise, $\square_{abcd}$ does not intersect S^* or it properly contains S^*, and in both cases it includes elements of V that need to be partitioned. We have also ensured at this point that $k \geq 1$. There are two options. One, we may leave $\square_{abcd}$ as one subset, but only if it is in $\mathcal{R}$ and does not contain S^*, and $k = 1$. In that case, by definition the percentile of S^* is $\frac{1.5}{2}$ if $\square_{abcd}$ is large, $\frac{1}{2}$ if medium, and $\frac{.5}{2}$ if small otherwise (if it is not possible, $\square_{abcd}$ is not a singleton and hence the second option will be possible). Two, we may subdivide $\square_{abcd}$ by cutting it vertically (assuming $a \neq b$) in one of $b - a$ positions or horizontally (assuming $c \neq d$) in one of $d - c$ positions and only further partitioning each of these subdivisions (no subsets crossing this cut). In that case, by direct computation the minimum percentile of S^* for $\square_{abcd}$ is the minimum (over all positions to make the cut and possible number of subsets in a partition of each adding up to k) convex combination computed by the algorithm of the minimum percentile of S^* for each resultant subgrid, since a hierarchical partition of $\square_{abcd}$ with a horizontal or vertical cut across which there may be no subsets is exactly comprised of a hierarchical partition of each subgrid. Finally, the minimum percentile of S^* for $\square_{abcd}$ is the minimum percentile resulting from each option. Hence if we suppose that COMPUTE-OPT(a', b', c', d', k') has the desired behavior for all $\square_{a'b'c'd'} \subsetneq \square_{abcd}$ and $k' \leq k$, the claim follows by induction on the size of the subgrid $(b-a)(d-c)$ and k (if the variable reaches 1 or 0, respectively, then no recursive calls are made).

To solve the grid with hierarchical rectangles variant of RANK PERCENTILE MAXIMIZATION,

we simply replace ∞ with $-\infty$ and min with max in Algorithm 4; the proof of correctness is analogous. □

3.4.3.2 Grid with Rectangles Variant

In the *grid with rectangles* variant, we consider the restriction where the only admissible partitions of V are those induced by partitioning the grid into rectangles with axis-parallel edges, not just ones arising from hierarchical partitioning. Additionally, each subset in the induced partition of V must be in $\mathcal{R}$.

Theorem 3.4.9. *The grid with rectangles variant of* Rank Maximization *is* NP*-hard.*

Proof. The special case of Maximum Disjoint Set where the collection of objects is comprised of axis-parallel rectangles in $\mathbb{R}^2$ is NP-hard [34], and this remains true if each rectangle consists of vertices with integer coordinates [12], so we fix such an instance of Maximum Disjoint Set. To convert this into an instance of Rank Maximization in the grid with rectangles variant, we proceed as follows.

- The grid dimensions are 2 times the maximum rectangle coordinate in the corresponding dimension
- For all $v \in V$, $\mu(\{v\}) = \frac{3}{5}\mu(S^*)$
- The vacancies are all grid points that have an odd coordinate
- $\mathcal{R}$ consists of the doubled version of each rectangle—i.e., the rectangle obtained by multiplying each of its vertex coordinates by 2—as delineated by the subset of vacancies inside its boundary

Clearly, two rectangles intersect if and only if their doubled versions intersect. By choice of μ, any rectangular subset $S \in \mathcal{R}$ of the doubled grid that contains at least two elements of V satisfies $\mu(S) \geq \mu(S^*)$, and any other subset does not. Hence, maximizing the rank of S^* in the Rank Maximization instance is equivalent to choosing a collection of disjoint rectangular subsets of maximum cardinality. To show that we only consider subsets identified as rectangles in the Maximum Disjoint Set instance, note that between any two grid points on the boundary of a rectangle in the Maximum Disjoint Set instance, a vacancy is placed between these points in its doubled version. As a result, the doubled version of each non-degenerate, rectangular subset in the original grid uniquely contains the subset of vacancies inside its boundary. □

Fortunately, the analogue with the better studied Maximum Disjoint Set yields an approximation algorithm for our problem.

Theorem 3.4.10. *For any $\epsilon > 0$, the grid with rectangles variant of* Rank Maximization *has a polynomial-time approximation algorithm with approximation guarantee* $\text{OPT} \leq (2 + \epsilon)\text{ALG} - (1 + \epsilon)$.

To see this, we leverage the fact that there is a $(2+\epsilon)$-approximation algorithm for the above special case of Maximum Disjoint Set [20]. The same guarantees apply to maximizing the number of medium or large subsets in the grid with rectangles variant of Rank Maximization because, conversely, this variant is a special case of Maximum Disjoint Set; the vacancies and $\mathcal{R}$ effectively just restrict the collection of rectangles we may consider.

We also have the following relationship between rank in these two variants, since a hierarchical partition is just a special class of partition.

Corollary 3.4.11. *Given an input (G, V_-, μ, S^*) for a grid variant problem, let* max rank *be the maximum rank in the grid with rectangles variant and let* max hierarchical rank *be the maximum rank in the grid with hierarchical rectangles variant; define* min rank *and* min hierarchical rank *analogously. Then* min rank $\leq$ min hierarchical rank $\leq$ max hierarchical rank $\leq$ max rank.

3.4.4 Applications of Recursive Algorithms for Rank and Percentile

The above recursive algorithms for the grid variants of the problems allow us to easily modify the criteria for subsets in a valid partitions and the convex combination of ranks or percentiles that we compute. We consider one such application below, then discuss another substantial one—which pertains to gerrymandering—in Chapter 4.

3.4.4.1 Numerical Grading over Multiple Periods

In several academic and professional contexts, including secondary schools in the United States, it is customary to evaluate people by recording a measure of average performance for a set number of terms and then computing a (possibly weighted) average of those marks. We consider an ex-post change in these terms to discuss the robustness of such grading practices.

Definition 3.4.12. *The* Weighted Average Maximization *problem is the following modification of the linear component case of* Rank Percentile Maximization*: we now*

require $S^* = \emptyset$. We are also given a measure μ^* such that $\mu \leq \mu^*$, and we drop the requirement that $\mu(S) = 0 \implies S = \emptyset$ but force $\mu^*(V) > 0$. The objective is now to select a valid partition $P = (S_1, S_2, \ldots, S_c)$, where $S_1 = [1, i_1], S_2 = [i_1 + 1, i_2], \ldots, S_c = [i_{c-1} + 1, i_c]$, $i_1 < i_2 < \cdots < i_{c-1} < i_c = l$, and $\mu^*(S_1) > 0, \mu^*(S_2) > 0, \ldots, \mu^*(S_c) > 0$ such that the grade $\frac{1}{i_1}\frac{\mu(S_1)}{\mu^*(S_1)} + \frac{1}{i_2 - i_1}\frac{\mu(S_2)}{\mu^*(S_2)} + \cdots + \frac{1}{i_c - i_{c-1}}\frac{\mu(S_c)}{\mu^*(S_c)}$ is maximized.

In this formulation, the horizontal dimension represents time, and each vertex represents a discrete time period, such as one day. The measure μ is the number of points earned for a particular interval of time, and μ^* is the maximum possible points for a particular interval of time. The goal is to partition the entire interval of time into marking periods such that the average percentage of points earned (weighted by the amount of time in each marking period) is as large as possible, yet manipulation by making some periods have zero points possible, effectively redistributing grading weight to other periods (including non-adjacent ones), is disallowed.

Because time is the only dimension of interest, we can obtain the below polynomial-time algorithm to solve this version by adapting the procedure for the linear component case of Rank Percentile Maximization.

Algorithm 5 Given an instance $(V, E, \mu^*, \mu, S^* = \emptyset)$ of Weighted Average Maximization, compute the maximum grade of a partition

for $1 \leq i \leq n$ **do**
  for $i \leq j \leq n$ **do**
    $$C[i,j] = \begin{cases} \frac{1}{j-i+1}\frac{\mu(\{v_i,\ldots,v_j\})}{\mu^*(\{v_i,\ldots,v_j\})} & \text{if } \mu^*(\{v_i,\ldots,v_j\}) > 0 \\ -\infty & \text{otherwise} \end{cases}$$
  end for
end for
$\mathrm{OPT}[1] = C[1,1]$
for $1 < j \leq n$ **do**
  $\mathrm{OPT}[j] = \sup_{1 \leq i < j}\{\mathrm{OPT}[i] + C[i+1, j]\}$
end for
return $\mathrm{OPT}[n]$

3.5 Numerical Experiments

The computational problems studied in this chapter were motivated by a variety of real-world contexts in which objects are ranked. In this section, we aim to apply the results

for these problems to compute the maximum and minimum rank and percentile of select objects of interest. A key point of concern is how robust the rank and percentile are to (perhaps manipulated) partitions of the ambient objects—for instance, is the maximum value close to the minimum?

We specialize our efforts to a salient topic: sources of greenhouse gas emissions, which is widely accepted by the scientific community as the leading cause of climate change. The relative contribution of sources to such emissions carries myriad implications for how blame is allocated, both to individual and corporate entities, and may shape policy debates on how to best mitigate climate change. The EPA itself ranks sources of greenhouse gas emissions at a coarse scale of categorization [38].

More specifically, we used the supplemental table "KCA-3: 2022 Key Category Approach 1 and Approach 2 Analysis—Level Assessment, without LULUCF" in the EPA's comprehensive report *Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990-2022*, which is intended to give an exhaustive breakdown of sources of emissions by CRT category, as introduced in Section 3.4.2, with the exception of "Land Use, Land-Use Change, and Forestry", a CRT category that the EPA sometimes omits due to the presence of intertwined greenhouse gas emission sources and sinks (whose separation could foment a form of manipulation in rankings that we are not studying in this chapter). These CRT codes allow us to model the categories according to the hierarchical category variant from Section 3.4.2, where we also create a category for each CRT Category that is listed for multiple greenhouse gases and append the corresponding greenhouse gas to the description. Our measure μ is the EPA estimate for the amount of emissions, in million metric tons of CO_2 equivalent, the underlying subset of activities caused in the year 2022.

The results indicate that large sources of emissions are somewhat robust in their placement in rankings. For instance, emissions due to road transportation (CRT Category: 1.A.3.b Transportation: Road) are always ranked first or second in any partition, and they always appear in the top 12% of ranked sources. More intermediate sources of emissions, however, experience much greater variability in ranking. Emissions due to cement production (CRT Category: 2.A.1 Cement Production (CO_2)) can be ranked as the 5th or 25th largest source of emissions and can appear after about 8% or 71% of other sources. Therefore, while factors such as automobiles may already dominate public discourse of personal responsibility and environmental regulations, there are a variety of other contributors to climate change whose relative importance may be manipulated by bad-faith actors and interpreted in widely

different ways arising from the same set of base facts.

The results for additional categories are shown in Table 3.1. The partitions achieving the indicated ranks and percentiles for 2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs are reproduced in Figures 3.3, 3.4, and 3.5. To optimize readability, node labels in these figures are factored hierarchically. Each parent node displays the maximal common string prefix of its subtree, while descendant nodes omit this inherited prefix to display only their unique differentiating suffixes. The full list of CRT categories used appears in Appendix A along with their absolute emission contributions.

CRT Category	**Min Rank**	**Max Rank**	**Min Percentile**	**Max Percentile**
1.A.3.b Transportation: Road	1	2	0.40%	11.54%
1.A.3.a Transportation: Aviation	7	13	7.39%	67.86%
2.A.1 Cement Production (CO_2)	5	25	8.46%	71.05%
2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs	3	13	4.72%	44.74%
3 Agriculture	2	6	2.34%	50.00%
3.B (Manure Management)	6	15	8.87%	71.88%

Table 3.1: Minimum and maximum rank and percentile (rounded to 2 decimal places) for select EPA Common Reporting Tables (CRT) categories over all valid hierarchical category partitions.

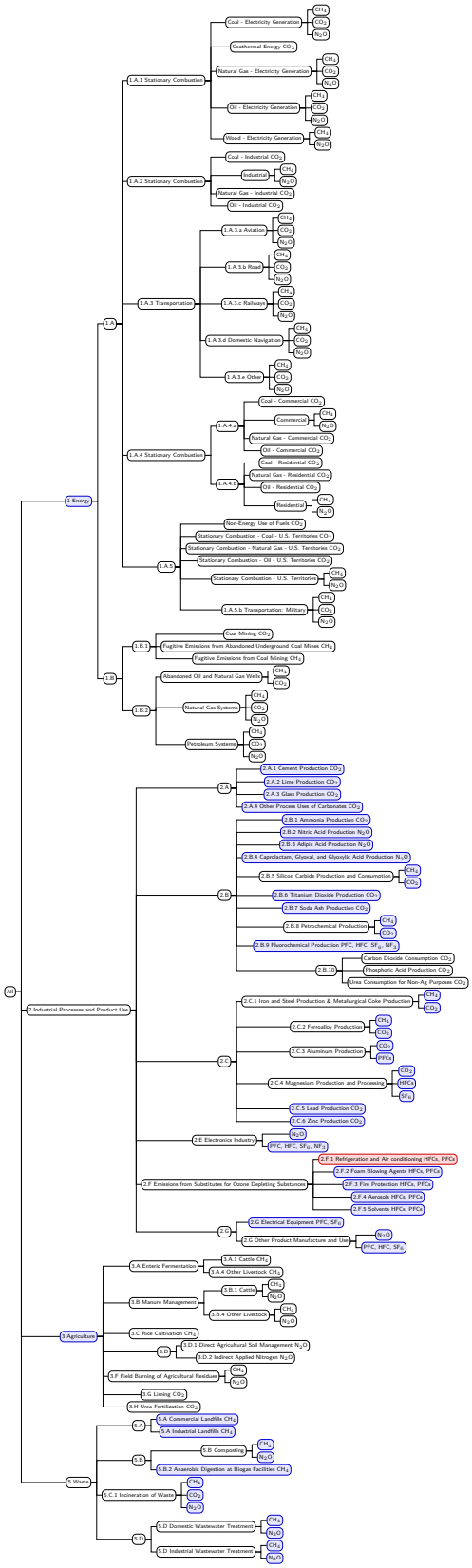


Figure 3.3: A partition that minimizes both the rank and rank percentile of 2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs, as recorded in Table 3.1. Subsets included in the partition are shaded.

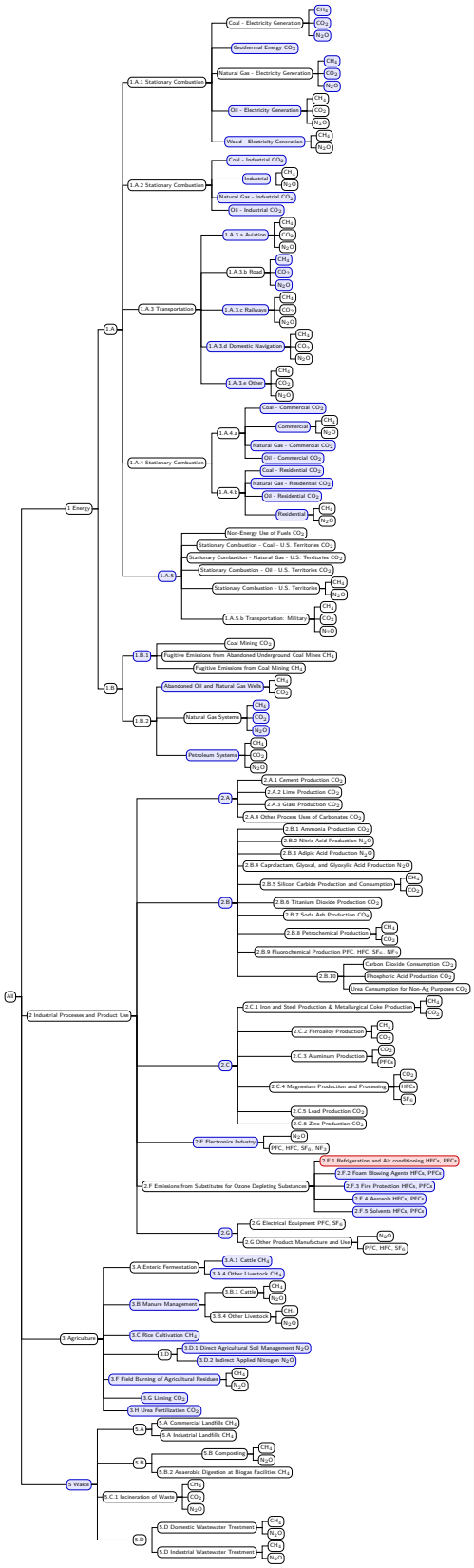


Figure 3.4: A partition that maximizes the rank of 2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs, as recorded in Table 3.1. Subsets included in the partition are shaded.

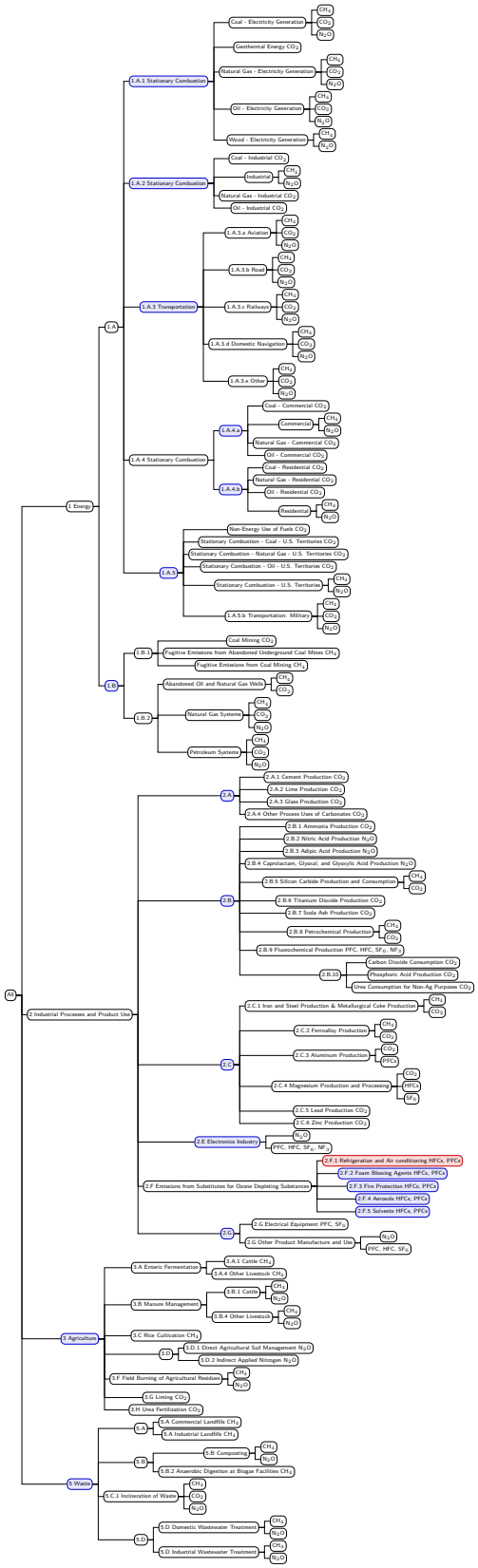


Figure 3.5: A partition that maximizes the rank percentile of 2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs, as recorded in Table 3.1. Subsets included in the partition are shaded.

CHAPTER 4

Redistricting with Hierarchical Rectangles

4.1 Introduction

Gerrymandering, which refers broadly to the strategic drawing of political boundaries to advantage a specific party, is a central challenge in representative democracies that elect representatives via winner-take-all voting in various geographic districts. While basic legal criteria such as contiguity—each district being connected—and population balance—each district applying to approximately the same number of people—are standard requirements for electoral maps, they are largely insufficient to prevent gerrymandering. One desideratum often discussed in representative democracy is proportional representation, where the number of seats allocated to a party is proportional to the total number of votes cast for it. While this consideration forms many electoral systems itself, we may also think of the extent to which it holds in a district-based system that may be legally prescribed or maintained for other benefits, such as the ability to organize political power with local contacts.

In this chapter, we formalize and analyze the problem of political redistricting on a grid, focusing specifically on partitions formed by axis-parallel rectangles. We progress from general contiguous districts to the more restrictive cases of arbitrary rectangle districts and, ultimately, hierarchical rectangle districts—partitions formed by recursively slicing a region with straight horizontal or vertical boundaries. By restricting the permissible shapes of electoral districts, we aim to understand how geometric rigidity affects a mapmaker's ability to maximize their party's representation, a problem we formally frame as Gerrymandering with Hierarchical Rectangles.

Our analysis establishes theoretical bounds on the discrepancy between proportional representation and the outcomes achievable under these different geometric constraints. We demonstrate that while hierarchical rectangular cuts severely restrict extreme packing and

cracking compared to arbitrary contiguous shapes, they can also structurally preclude proportional representation (as illustrated by our "pinwheel" construction). Furthermore, we show that unlike many general redistricting problems that are computationally intractable, the Gerrymandering with Hierarchical Rectangles problem can be solved exactly in polynomial time utilizing a dynamic programming approach.

Finally, we extend our general framework in two directions to more closely model real-world dynamics. First, we introduce the Redistricting Game with Hierarchical Rectangles, a protocol where competing political actors alternate drawing hierarchical boundaries. Second, we propose an urban-rural model characterized by a linear decay of partisan support moving away from an urban center to evaluate whether realistic demographic distributions naturally mitigate the gerrymandering capacity of hierarchical rectangles, the answer being a qualified yes. Our results thus offer a duality in which the more restrictive gerrymandering model is precluded from realizing proportional representation in certain contexts and restricted to proportional representation in others.

4.2 Contiguous Redistricting and Proportional Representation

In this section, we augment the general set-up of our four partitioning problems in Chapter 3, which required a vertex set V, a binary relation E on $V \times V$ (which determines a set of connected vertices $\mathcal{S}$), a measure μ on 2^V, and a subset S^* of V. Now, we are also given $N \in \mathbb{N}$, a signed measure μ_R on 2^V, and a parameter $\rho \in [0, \frac{1}{N+1})$ such that $|\mu_R| \leq \mu$, each valid partition must also have exactly N subsets, and any subset $S \in \mathcal{S}$ in a valid partition must also satisfy $(1-\rho)\frac{\mu(V)}{N} \leq \mu(S) \leq (1+\rho)\frac{\mu(V)}{N}$.

We may now interpret the parameters as follows. A state, which is subdivided into the elements of V (which we call *precincts*) and whose adjacency is represented by E, is allocated N districts in a legislative body, each of which should be contiguous and have population (measured by μ) within a fraction ρ of the eventual average district population for that state. Two parties are ascendant in the political system, labeled Player 1 and Player 2, and μ_R is the number of Player 1-leaning voters minus the number of Player 2-leaning voters (or the expectation thereof) for a given area, so by this convention the objective for this version of Rank Maximization would be to choose a redistricting plan satisfying the above rules such that the number of districts that lean toward Player 1 by at least the margin of S^* is maximized. For concreteness, we study the special case where $S^* = \emptyset$; we

call this variant of rank maximization GERRYMANDERING WITH CONTIGUOUS DISTRICTS.

More formally, we have the following definitions.

Definition 4.2.1 (Valid Contiguous Redistricting, District, and Slate)**.** *Given a tuple $(V, E, \mu, \mu_R, \rho, N)$ as above, we say a partition $(S_1, \ldots, S_N)$ of V is a valid contiguous redistricting if $S_1, \ldots, S_N \in \mathcal{S}$ and $(1-\rho)\frac{\mu(V)}{N} \leq \mu(S_i) \leq (1+\rho)\frac{\mu(V)}{N}$ for all $i \in \{1, \ldots, N\}$, and each subset S_i is called a district thereof. Given a valid contiguous redistricting $(S_1, \ldots, S_N)$ and $j \in \{1, 2\}$, we say Player 1's slate N_1 is the number of S_i for which $\mu_R(S_i) \geq 0$ and Player 2's slate N_2 is the number of S_i for which $\mu_R(S_i) < 0$ (i.e., ties are broken in favor of Player 1).*

Definition 4.2.2 (Gerrymandering with Contiguous Districts)**.** *The* GERRYMANDERING WITH CONTIGUOUS DISTRICTS *problem is as follows: given a tuple $(V, E, \mu, \mu_R, \rho, N)$ as above, select a valid contiguous redistricting $(S_1, \ldots, S_N)$ that maximizes the number of S_i for which $\mu_R(S_i) \geq 0$.*

Because we are only interested in *differences* in votes between two parties, we may assign each member of the population a loyalty fraction in $[0, 1]$ representing the probability they will vote for Player $j \in \{1, 2\}$ (using, for instance, a value of .5 for those ineligible to vote or voters with minimal turnout in previous elections), take the average loyalty fraction in a precinct v over its $\mu(\{v\})$ people to obtain a precinct loyalty fraction, and take the average loyalty fraction over all people (respectively all people in a subset of V) to get a total loyalty fraction p_j (respectively subset loyalty fraction). Proportional representation concerns how close the number of districts N_j given to Player j in a valid contiguous redistricting as above is to $p_j N$, which is the number of districts Player j would receive if districts were awarded to parties proportional to their votes. (Here, while the placeholder value of .5 does not affect the winner of any prospective district, it does make p_1 and p_2 closer to .5 themselves, which may be reasonable to the extent that non-voting reflects indifference to the two alternatives, but less so when it reflects barriers to voting or ineligibility. Nonetheless, such issues are often disregarded in modeling assumptions, cf. [30],[5].)

We present the following standard inequality comparing N_j in general to quantities related to proportional representation.

Theorem 4.2.3. *In any valid contiguous redistricting,*

$$\left\lceil \frac{2(p_j - .5 - .5\rho)N}{1-\rho} \right\rceil \leq N_j \leq \left\lfloor \frac{2p_j N}{1-\rho} \right\rfloor.$$

Proof. Fix $j \in \{1, 2\}$. First suppose $\rho = 0$. Because Player j needs a subset loyalty fraction of at least .5 to win each corresponding subset N_i, we have that $.5N_j$ is at least the total loyalty fraction for Player j, i.e. $.5N_j \leq p_j N$, so $N_j \leq 2p_j N$. Analogously, for the other Player $-j$, $N_{-j} \leq 2p_{-j}N$, which we combine with the fact that $p_j + p_{-j} = 1$ to obtain $2(p_j - .5)N \leq N_j$. If instead $\rho \neq 0$, the districts loyal to Player j may be the least populous, yielding the inequality $.5N_j(1 - \rho) \leq p_j N$, so $N_j \leq \frac{2p_j N}{1-\rho}$, and we analogously obtain the desired inequality. □

The proof also indicates when these bounds on N_j are achievable; namely, when one player has bifurcated support in which their subset loyalty fraction is just at or above .5 for some districts and close to 0 for the remaining districts. We now explore the extent to which this is the case when we restrict to districts that arise from rectangular partitioning as well as hierarchical, axis-parallel partitioning.

4.3 Redistricting with (Hierarchical) Rectangles

We now adapt grid variants of the partitioning problems, which will give us additional structure for analyzing redistricting. We impose the following new conditions: $V = ([1, l] \times [1, w])\backslash V_-$, where $V_- \subset [1, l] \times [1, w]$ represents a set of vacancies and $n = |V| = lw - |V_-|$; $((a, c), (b, d)) \in E \iff |b - a| + |d - c| = 1$; and the set of rectangles that may be in a valid partition, $\mathcal{R}$, satisfies $\square_{abcd} \cap V_- = \emptyset \implies \square_{abcd} \in \mathcal{R}$ and $\square_{abcd} \cap V_- = \square_{a'b'c'd'} \cap V_- \implies (\square_{abcd} \in \mathcal{R} \iff \square_{a'b'c'd'} \in \mathcal{R})$. The introduction of V_- allows us to respect geographic and other features while applying our results to states that may not be rectangular on their own by omitting points.

If we simply require each set in a valid partition to be a rectangle, we obtain the rectangles variant of Gerrymandering with Contiguous Districts. If the only admissible partitions of V are those induced by hierarchically partitioning the grid into rectangles with axis-parallel edges, we obtain the hierarchical rectangles variant of Gerrymandering with Contiguous Districts.

A formal statement of these models, and their relation to each other, is included below.

Definition 4.3.1 (Valid Rectangle Redistricting)**.** *Given a tuple $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ as above, we say a partition $(S_1, \ldots, S_N)$ of V is a valid rectangle redistricting if it is a valid contiguous redistricting and for each $i \in \{1, \ldots, N\}$, $S_i \in \mathcal{R}$.*

Definition 4.3.2 (Gerrymandering with Rectangles)**.** *The* GERRYMANDERING WITH RECTANGLES *problem is as follows: given a tuple* $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ *as above, select a valid rectangle redistricting* $(S_1, \ldots, S_N)$ *that maximizes the number of* S_i *for which* $\mu_R(S_i) \geq 0$*.*

Definition 4.3.3 (Valid Hierarchical Rectangle Redistricting)**.** *Given a tuple* $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ *as above, we say a partition* $(S_1, \ldots, S_N)$ *of* V *is a valid hierarchical rectangle redistricting if it is a valid rectangle redistricting and is induced by hierarchically partitioning the grid into rectangles with axis-parallel edges. That is, in each step, we take a subgrid from a previous step (*$[1, l] \times [1, w]$ *for the first step) and draw a horizontal or vertical line between two rows or columns thereof, respectively, resulting in the cut subgrid becoming two subgrids separated by the line instead.*

Definition 4.3.4 (Gerrymandering with Hierarchical Rectangles)**.** *The* GERRYMANDERING WITH HIERARCHICAL RECTANGLES *problem is as follows: given a tuple* $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ *as above, select a valid hierarchical rectangle redistricting* $(S_1, \ldots, S_N)$ *that maximizes the number of* S_i *for which* $\mu_R(S_i) \geq 0$*.*

The use of hierarchical cuts seems quite restrictive a priori, and raises the specter of reaching a state from which no feasible cuts may be performed to complete the partition. Precluding such phenomena provides our initial motivation for the following definition.

Definition 4.3.5 (Population Regularity)**.** *Given* $V = [1, l] \times [1, w] \subset \mathbb{N}^2$*, denote by* $V_{i,\cdot} = \{v \in V : \exists j \mid v = (i, j)\}$ *and* $V_{\cdot,j} = \{v \in V : \exists i \mid v = (i, j)\}$ *the rows and columns of* V*, respectively. We say an instance* $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ *of* GERRYMANDERING WITH RECTANGLES *satisfies population regularity (PR) if for all* $i \in [1, l]$ *and* $j \in [1, w]$*,* $\mu(V_{i,\cdot}) \leq \rho \frac{\mu(V)}{N}$ *and* $\mu(V_{\cdot,j}) \leq \rho \frac{\mu(V)}{N}$*.*

Population regularity states that each row or column (hence any portion thereof) has population no more than the fraction ρ of the mean population for a district in an eventual redistricting, which is the maximum difference a district's population can deviate from the mean population for a district. As a result, whenever we need to make a cut in V (or a subgrid thereof), we can achieve any allocation of its prescribed number of districts to the left of and to the right of or above and below the next cut, but this requires some analysis of the propagation of such differences from the mean population through repeated cuts. Indeed, to see that we may not increase the upper bound on each row and column population by a factor of 2 in the definition for (PR), only ensuring that each district we form has population within ρ of $\frac{\mu(V)}{N}$, suppose $N = 3$ and $\mu(V) = 3$ and that we have formed one district with population $1 - \rho$, leaving a subgrid V' with $\mu(V') = 2 + \rho$ that must be partitioned into 2

districts. Then, there may exist j^* such that the population of the first j^* columns within V_- is $1-\frac{\rho}{2}$ (so the population of the remaining columns within V_- is $1+\frac{3\rho}{2}$) but the population of the first j^*+1 columns within V_- is $1+\frac{3\rho}{2}$ (so the population of the remaining columns within V_- is $1-\frac{\rho}{2}$). In this case, there is no valid vertical cut that will complete a valid hierarchical rectangle redistricting of V.

Lemma 4.3.6. *Let $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ be an instance of* Gerrymandering with Hierarchical Rectangles *satisfying (PR) and suppose V' is a subgrid of V with population in the interval $[\frac{\mu(V)}{N}N_0-\rho, \frac{\mu(V)}{N}N_0+\rho]$ for some $N_0 \in \{1,2,\dots\}$. First suppose $N_0 > 1$. Let $N^* \in \{1,\dots,N_0-1\}$. Then there exist: $i^* \in [1, l-1]$ such that the horizontal cut restricted to V' between rows $V'_{i^*,\cdot}$ and $V'_{i^*+1,\cdot}$ is the first cut within V' in a valid hierarchical rectangle redistricting of V' into N^* districts arising from partitioning $V'_{1,\cdot},\dots,V'_{i^*,\cdot}$ and $N-N^*$ districts arising from partitioning of $V'_{i^*+1,\cdot},\dots,V'_{l,\cdot}$ (except that each set has population in $[\frac{\mu(V)}{N}(1-\rho), \frac{\mu(V)}{N}(1+\rho)]$); and $j^* \in [1, w-1]$ such that the vertical cut restricted to V' between columns $V'_{\cdot,j^*}$ and $V'_{\cdot,j^*+1}$ is the first cut within V' in a valid hierarchical rectangle redistricting of V' with N^* districts arising from partitioning $V'_{\cdot,1},\dots,V'_{\cdot,j^*}$ and $N-N^*$ districts arising from partitioning $V'_{\cdot,j^*+1},\dots,V'_{\cdot,w}$ (except that each set has population in $[\frac{\mu(V)}{N}(1-\rho), \frac{\mu(V)}{N}(1+\rho)]$). If instead $N_0 = 1$, then there exists a valid hierarchical rectangle redistricting of V' as one set with population in $[\frac{\mu(V)}{N}(1-\rho), \frac{\mu(V)}{N}(1+\rho)]$.*

Proof. Fix an instance $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ of Gerrymandering with Hierarchical Rectangles satisfying (PR) as in the lemma. We prove the statement for all V' as in the lemma by induction on N_0 (denote by r and c the number of rows and columns V' has, respectively). For $N_0 = 1$, we have a rectangular subgrid V' with population in the interval $[\frac{\mu(V)}{N}-\rho, \frac{\mu(V)}{N}+\rho]$, hence constituting a rectangle district of the requisite population. Now suppose the statement is true for all V' with $N_0 \le M_0 - 1$ and choose a V' with $N_0 = M_0$ and some $N^* \in \{1,\dots,M_0-1\}$. Without loss of generality $\mu(V') = M_0 \cdot \frac{\mu(V)}{N} + \epsilon$, where $0 \le \epsilon \le \rho$. Consider the quantity $N^* \cdot \frac{\mu(V)}{N} + \frac{\rho}{2}$ and note that $\mu(\emptyset) = 0$ whereas $\mu(V'_{1,\cdot}) + \mu(V'_{2,\cdot}) + \cdots + \mu(V'_{r,\cdot}) = \mu(V') = \mu(V'_{\cdot,1}) + \mu(V'_{\cdot,2}) + \cdots + \mu(V'_{\cdot,c})$. Thus by (PR), there exist $i^* \in [1, r-1]$ such that $\mu(V'_{1,\cdot}) + \mu(V'_{2,\cdot}) + \cdots + \mu(V'_{i^*,\cdot})$ is within $\frac{\rho}{2}$ of $N^* \cdot \frac{\mu(V)}{N} + \frac{\rho}{2}$ and $j^* \in [1, c-1]$ such that $\mu(V'_{\cdot,1}) + \mu(V'_{\cdot,2}) + \cdots + \mu(V'_{\cdot,c^*})$ is within $\frac{\rho}{2}$ of $N^* \cdot \frac{\mu(V)}{N} + \frac{\rho}{2}$. As a result, $\mu(V'_{1,\cdot}) + \mu(V'_{2,\cdot}) + \cdots + \mu(V'_{i^*,\cdot}) \in [\frac{\mu(V)}{N} \cdot N^*, \frac{\mu(V)}{N} \cdot N^* + \rho]$, so $\mu(V'_{i^*+1,\cdot}) + \mu(V'_{i^*+2,\cdot}) + \cdots + \mu(V'_{r,\cdot}) \in [\frac{\mu(V)}{N} \cdot (M_0 - N^*) + \epsilon - \rho, \frac{\mu(V)}{N} \cdot (M_0 - N^*) + \epsilon]$. Now applying the inductive hypothesis with $N_0 = N^*, M_0 - N^*$ and using the fact that a partition of $V'_{1,\cdot},\dots,V'_{i^*,\cdot}$ and a partition of $V'_{i^*+1,\cdot},\dots,V'_{l,\cdot}$ form a partition of V' where the first cut within V' is the horizontal cut between rows $V'_{i^*,\cdot}$ and $V'_{i^*+1,\cdot}$, i^* is of the form as in the lemma. Similarly, j^* is of the form as in the lemma. □

Taking $V' = V$ gives the desired ability to partition the grid using hierarchical cuts while satisfying the population requirements.

4.3.1 Worst-Case Gaps with Proportional Representation

Proportional representation is frequently regarded as a desideratum for legislative bodies, and while geographical districts still hold advantages (such as increased ability for community organization), it is valuable to assess how well their representatives might reflect the entire state's preferences in a system of non-local political parties, as is the case in many countries. Each successive redistricting rule set stated above limits the types of districts that are allowed, which limits the set of possible slates a player may achieve from redistricting. This prompts us to study the geometry excluded in each case, the implications for proportional representation and gerrymandering, and likely manifestations in the real-world.

The following structured cases represent scenarios where proportional representation fails to the greatest possible extent under Theorem 4.2.3.

The first example illustrates this disparity for rectangle redistricting compared to contiguous redistricting, even in the presence of population regularity.

Theorem 4.3.7. *For all $N \in \mathbb{N}$, there exists an instance $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ of* Gerrymandering with Rectangles *satisfying (PR) and $\mathcal{R}^c = \emptyset$ such that some valid contiguous redistricting satisfies $N_1 = \lfloor p_1 N \rceil$ but all valid rectangle redistrictings satisfy $N_1 = \left\lceil \frac{2(p_1 - .5 - .5\rho)N}{1-\rho} \right\rceil$.*

Proof. Let S^* and $V_-, \mathcal{R}^c = \emptyset$. Define an auxiliary grid V_0 to consist of $\lfloor \frac{N}{4} \rfloor$ 4×4 subgrids, where we designate one to be placed first; any other subgrid is placed to the right of or below the first one or an already placed subgrid. If $N \bmod 4 \in \{2, 3\}$, we also place a 2×2 subgrid to the right or below the last 4×4 subgrid placed (aligned with its 2 leftmost columns or 2 topmost rows, respectively), and if $N \bmod 4 \in \{1, 3\}$, we also place a column to the right or a row below the rest of V_0 (such that it remains a rectangular grid). Set $\mu(V) = N$. Now for each subgrid, we give their diagonal entries population 5, their off-diagonal entries in the 2 leftmost columns population 3 below the diagonal and population 5 above the diagonal, and their off-diagonal entries in the 2 rightmost columns (if the subgrid is 4×4) population 5 below the diagonal and population 3 above the diagonal (under μ). In each subgrid, each diagonal entry has precinct loyalty fraction $.5 + \delta$, and each off-diagonal entry has precinct loyalty fraction $.5 - 2\delta$, where $0 < \delta \leq .25$. In the case $N \bmod 4 \in \{1, 3\}$, the column or row

added has population 15 split equally between its entries, all of which have precinct loyalty fraction $.5 - \delta$. Lastly, we form V by replacing each entry of V_0 with a $T \times T$ subgrid, where $T = \frac{\max_{i,j}\{V_{i,\cdot}, V_{\cdot,j}\}}{2\rho\mu(V_0)/N}$, the precinct loyalty fractions are unchanged, and the precinct populations are divided by T^2. The result is that $\mu(V) = \mu(V_0)$ and population regularity is satisfied—we can take "fragments" of our original precincts in V_0 (still inheriting the same level of support for Player 1) to form our partition. Also note that $p_1 = .5 - \delta$. An example of the construction is shown in Figure 4.1.

Our valid contiguous redistricting consists of the $T \times T$ subgrids for the following positions of V_0: the 2 diagonal entries in the 2 leftmost columns with the entry of population 5 above one of them from each subgrid; the 5 off-diagonal entries in the 2 leftmost columns with population 3 from each subgrid; the 2 diagonal entries in the 2 rightmost columns with the entry of population 5 below one of them from each 4×4 subgrid; and the 5 off-diagonal entries in the 2 rightmost columns with population 3 from each 4×4 subgrid. By inspection, Player 1's slate is $\lfloor \frac{N}{2} \rfloor$. As $\delta \downarrow 0$, $p_1 N \uparrow \frac{N}{2}$, so $N_1 = \lfloor p_1 N \rceil$ for δ sufficiently small.

On the other hand, any valid rectangle redistricting gives Player 1 a slate of 0. To see this, note that in V, any rectangle satisfying the district population requirements and loyal to Player 1 must enclose more than 1 diagonal entry (when adding all portions), hence enclosing at least 2 times as many off-diagonal entries (when adding all portions), so the fraction of population of any district arising from diagonal entries is at most $\frac{2 \cdot 5}{2 \cdot 5 + 2 \cdot 3} = \frac{10}{16} < \frac{2}{3}$, meaning its loyalty fraction for Player 1 is less than .5. As a result, Player 1's slate must be 0 (regardless of the value of ρ), and for δ and ρ sufficiently small this corresponds to the general lower bound for N_1. □

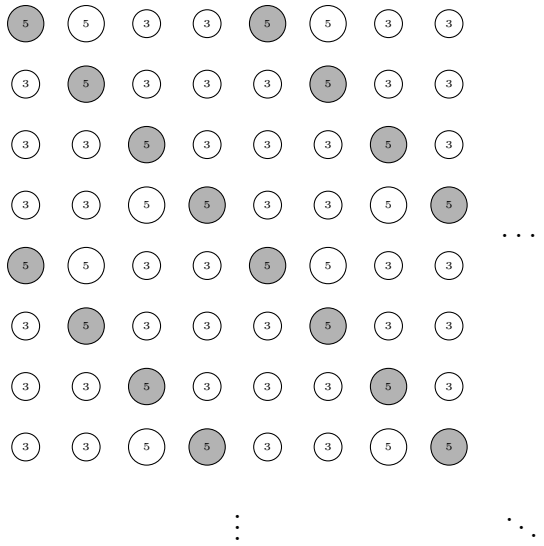


Figure 4.1: The construction of V for Theorem 4.3.7 with $N \geq 16$ (at least four 4×4 subgrids), where $T = 1$. Shaded circles indicate Player 1's loyalty fraction is greater than 0.5; circle area is proportional to population. For values of N larger than 16, the pattern extends, as denoted by the ellipses.

If we instead choose $\delta = .25$ for the construction in Theorem 4.3.7, then the results are more mixed. We obtain $p_1 = .25$, so Player 1's maximum slates from each of the models are approximately equidistant to the slate Player 1 would have under proportional representation. Restricting to rectangle districts then results in Player 1 not receiving any representation, whereas allowing any contiguous districts results in Player 1 being roughly tied for districts with Player 2 even though the latter receives 3 times the vote share.

We now explore a similar disparity for hierarchical rectangle redistricting compared to rectangle redistricting.

Theorem 4.3.8. *For all $N \in \mathbb{N}$: $N \bmod 6 = 0$, there exists an instance $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ of* Gerrymandering with Hierarchical Rectangles *satisfying (PR) such that some valid rectangle redistricting satisfies $N_1 = \lfloor p_1 N \rceil$ but all valid hierarchical rectangle redistrictings satisfy $N_1 = \left\lceil \frac{2(p_1 - .5 - .5\rho)N}{1-\rho} \right\rceil$.*

Proof. To see this, we employ the classical "pinwheel" construction, a minimal example of 5 rectangles comprising the grid that may not be obtained via hierarchical partitioning [35, 8]. Define a gadget grid $V_g = [1, 8] \times [1, 8]$ with the following rectangles: $R_1 = [1, 6] \times [7, 8]$, $R_2 = [1, 2] \times [1, 6]$, $R_3 = [3, 8] \times [1, 2]$, $R_4 = [7, 8] \times [3, 8]$, and $R_5 = [3, 6] \times [3, 6]$. We also define vacancies $V_-^0 = ([9, 9] \times [1, 8]) \cup ([18, 18] \times [1, 8])$ and a dummy grid $V_d = [10, 17] \times [1, 8]$. The rectangles R_1, R_3 and R_5 have population $1 - \rho$, respectively, whereas the rectangles

R_2, R_4, and V_d have population $1+\rho$. The portion of R_1 above R_2, of R_2 to the left of R_3, of R_3 below R_4, and of R_4 to the right of R_1 have population $.2(1-\rho)$, $.2(1+\rho)$, $.2(1-\rho)$, and $.2(1-\rho)$, respectively, but for each R_i within these portions and their complements the population is uniform over precincts. The precinct loyalty fractions are as follows:

- $(1,8),(8,1)$: p_1
- $(3,4),(6,5)$: p_2
- $R_1\backslash\{(1,8)\}, R_3\backslash\{(8,1)\}$: n_1
- $R_5\backslash\{(3,4)\cup(6,5)\}$: n_2
- $R_2\cup R_4\cup V_d$: n_3

where $p_1=.5+\delta$, $n_1=.5-\frac{\delta}{11}$, $p_2=.5+\frac{\delta}{2}$, $n_2=.5-\frac{\delta}{28}$, and $n_3=.5-\frac{3\delta}{2}$.

To form V and V_-, we take the union of $\frac{N}{6}$ copies of $V_g\cup V_d$ and V_-^0, respectively with the jth copy, $j\in\{2,3,\ldots,\frac{N}{6}\}$ shifted horizontally by $18j$, including all populations and precinct loyalty fractions, then amplify each precinct b by some $T\times T$ subgrid as in the proof of Theorem 4.3.7 to achieve (PR) (still referring to such a $T\times T$ subgrid by the original coordinates of the initial precinct and saying "taking a portion" of them to mean some but not all of the precincts in the new $T\times T$ subgrid). The set $\mathcal{R}$ consists of all rectangles that have empty intersection with V_-. The construction is depicted in Figure 4.2.

We first argue that the rectangles that may contribute to Player 1's slate in a valid rectangle redistricting are precisely R_1, R_3, R_5, and their horizontal translates. By choice of p_1, p_2, n_1, and n_2, each of these rectangles has loyalty fraction .5, so it remains to rule out any other rectangle. Each of these rectangles also has population $1-\rho$, so any proper subset of one may not form a district. Moreover, each copy of V_g is necessarily partitioned into exactly 5 rectangles—without any other precincts—due to the fact that districts may not contain any element of V_-, a set whose columns delineate each copy of V_g from the adjacent copies of V_d. Thus we now rule out any rectangle in V_g that overlaps multiple R_i. A district encompassing portions of $(1,8)$ and $(8,1)$ would have population at least $(1-\rho)(1+2\cdot.8)=2.6$, which is not allowed. A district encompassing portions of $(1,8)$ and $(3,4),(6,5)$ would enclose 2 precincts with loyalty fraction n_3 and at least the same fraction of such a precinct as is taken of $(3,4)$, which will give loyalty fraction less than .5 because $|n_3-.5|=|p_1-.5|+|p_2-.5|$. Analogous arguments apply with $(8,1)$ instead of $(1,8)$. Therefore, no winning district may have portions of precincts with loyalty greater than .5 that are from distinct R_i.

Also, only R_2 may enclose $(3,4)$ and $(6,5)$ because its loyalty fraction is exactly .5 and $|n_2 - .5| < |n_1 - .5|, |n_3 - .5|$. Only R_1 and R_3 may enclose $(1,8)$ and $(8,1)$, respectively, per our analysis involving precincts from R_2 (precluding any with precinct loyalty fraction greater than .5 only makes things worse because the precincts with loyalty fraction n_3 used in the above analysis are always traversed even if fewer precincts with loyalty fraction n_2 are.

Next, we argue that any valid hierarchical rectangle redistricting may not contain any such rectangle. Recall that each copy of V_g is necessarily partitioned into exactly 5 rectangles, which prescribes exactly 4 hierarchical cuts, and we may analyze each copy of V_g separately. Suppose there exists a valid hierarchical rectangle redistricting that contains R_1 as a subset. Consider the subgrid that is below the upper left district, which has population $2.8(1 \pm \rho)$. Therefore, 2 cuts must pass through the subgrid, whereas the remaining 2 cuts would need to be along the bottom (horizontal) and right (vertical) sides of R_1. To avoid violating aforementioned population constraints, the first cut made must be one that passes through the subgrid, and to avoid splitting R_1, it must be horizontal. The second cut cannot be on the horizontal side of R_1 (that would separate a region with population $1.2 \pm \rho$) nor on the vertical side of R_1 (the induced uncut upper right region would have population deviating from 1 by more than ρ), so the second cut must be through the subgrid, and at least one of these two cuts is horizontal and above the horizontal side of R_3 (if the second cut is horizontal, both of them cannot be below the horizontal side of R_3; if the second cut is vertical, then the first cut has to be above the horizontal side of R_3 in order to avoid splitting the upper left district or splitting a region with population $1 \pm \rho$). Moreover, one of the two cuts must lie above R_3 to avoid violating population constraints. The third cut must be along the vertical side of R_1 (the horizontal side cut would separate a region with population $1.2 \pm \rho$), causing the upper right region of the grid to have population less than $1 - \rho$. This is a contradiction, so there is no such valid hierarchical rectangle redistricting. By symmetry, the corresponding statement applies to R_3. As for R_5, because only 4 cuts are made within V_g, the first cut must be along one of the sides of R_5, creating a region with population $1.2 \pm \rho$, which prevents a valid hierarchical rectangle redistricting from being completed.

Hence any valid hierarchical rectangle redistricting has a slate for Player 1 of 0, but a valid rectangle redistricting may consists of $R_1, R_2, R_3, R_4, R_5, V_g$ and their horizontal translates, yielding a slate for Player 1 of $\frac{N}{2}$. For δ and ρ sufficiently small, these quantities align with the expressions in the theorem statement. $\square$

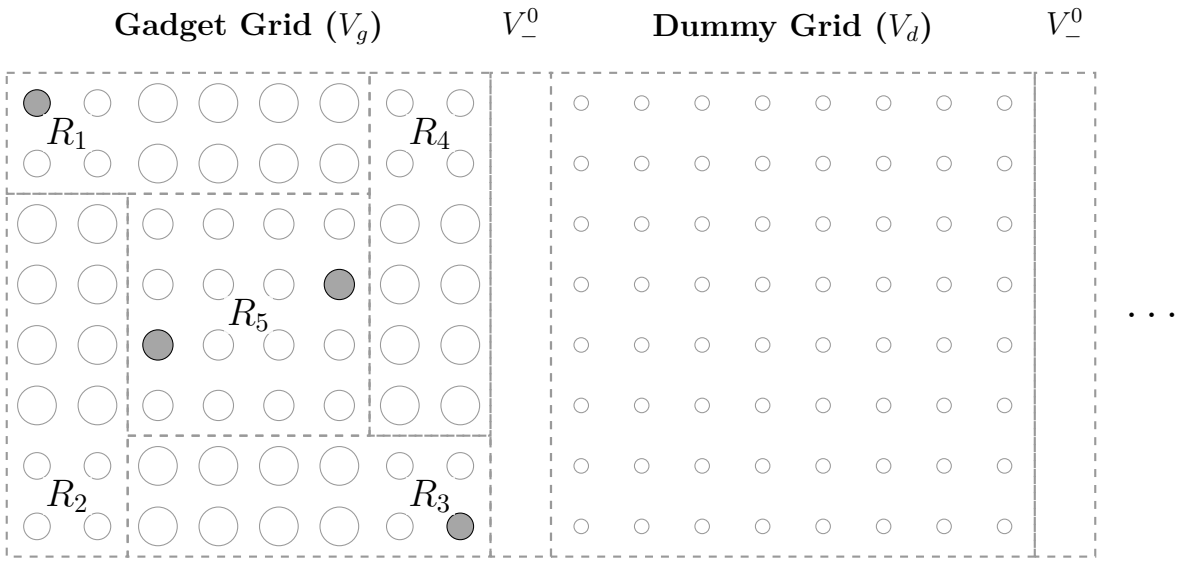


Figure 4.2: The construction of V for Theorem 4.3.8 with $N \geq 6$, where $T = 1$. Shaded circles indicate Player 1's loyalty fraction is greater than 0.5; circle area is proportional to population. For values of N larger than 6, the pattern extends, as denoted by the ellipses.

4.3.2 Solving Gerrymandering with Hierarchical Rectangles

In Chapter 3, we briefly touched on applications of our recursive approach to optimizing percentile within a grid. Here, we apply it to actually solve the hierarchical rectangles variant of Gerrymandering with Contiguous Districts.

There are two main impediments to using the techniques in Chapter 3 to study gerrymandering: monotonicity of unsigned measures is frequently leveraged and subsets in a valid partition are not required to be a similar size (even if this outcome is likely in the maximization problems). We circumvent both by adapting the algorithm for Rank Maximization in the grid with hierarchical rectangles variant, restricting to districts of this form in order to check the new redistricting conditions in our recursive solution approach.

A polynomial-time algorithm solving Gerrymandering with Hierarchical Rectangles by modifying the procedure for the grid with hierarchical rectangles variant of Rank Maximization is presented below.

Theorem 4.3.9. Gerrymandering with Hierarchical Rectangles *is polynomial-time solvable.*

Proof. Consider the following algorithm.

Algorithm 6 Given an instance $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ of GERRYMANDERING WITH HIERARCHICAL RECTANGLES, compute the maximum slate of Player 1

Initialize four-dimensional table OPT of size $l \times l \times w \times w$ with NULL entries
COMPUTE-OPT$(1, l, 1, w)$

procedure COMPUTE-OPT(a, b, c, d)
  if OPT$[a, b, c, d]$ is not NULL **then**
    return OPT$[a, b, c, d]$
  else if $\square_{abcd} \subset V_-$ **then**
    OPT$[a, b, c, d] = 0$
  end if
  $$c_b \leftarrow \begin{cases} 1 & \text{if } \mu_R(\square_{abcd}) \geq 0,\ (1-\rho)\frac{\mu(V)}{N} \leq \mu(\square_{abcd}) \leq (1+\rho)\frac{\mu(V)}{N}, \text{ and } \square_{abcd} \in \mathcal{R} \\ 0 & \text{if } \mu_R(\square_{abcd}) < 0,\ (1-\rho)\frac{\mu(V)}{N} \leq \mu(\square_{abcd}) \leq (1+\rho)\frac{\mu(V)}{N}, \text{ and } \square_{abcd} \in \mathcal{R} \\ -\infty & \text{otherwise} \end{cases}$$
  $c_v \leftarrow -\infty$
  $c_h \leftarrow -\infty$
  if $a < b$ **then**
    $c_v \leftarrow \max_{a \leq i < b} \{$COMPUTE-OPT$(a, i, c, d)$ + COMPUTE-OPT$(i+1, b, c, d)\}$
  end if
  if $c < d$ **then**
    $c_h \leftarrow \max_{c \leq j < d} \{$COMPUTE-OPT$(a, b, c, j)$ + COMPUTE-OPT$(a, b, j+1, d)\}$
  end if
  OPT$[a, b, c, d] = \max\{c_b, c_v, c_h\}$
  return OPT$[a, b, c, d]$
end procedure

We claim that for all $1 \leq a \leq b \leq l, 1 \leq c \leq d \leq w$, COMPUTE-OPT$(a, b, c, d)$ gives the maximum number of rectangles with μ_R non-negative when $\square_{abcd} \backslash V_-$ is hierarchically partitioned into districts D in $\mathcal{R}$ with the further restriction that for each D, $(1-\rho)\frac{\mu(V)}{N} \leq \mu(D) \leq (1+\rho)\frac{\mu(V)}{N}$ (unless such a partition is not possible, in which case COMPUTE-OPT(a, b, c, d) should be negative infinity). Taking $a = 1, b = l, c = 1, d = w$ will then prove the correctness of the algorithm because if V is hierarchically partitioned in this manner, by construction the result is a valid hierarchical rectangle redistricting with exactly N districts. Indeed, since $\rho < \frac{1}{N+1}$, $N_- < N$ districts have population at most $(1+\rho)\frac{\mu(V)}{N} N_- < \mu(V)$ and $N_+ > N$ districts have population at least $(1-\rho)\frac{\mu(V)}{N} N_+ > \mu(V)$.

First, if $\square_{abcd} \subset V_-$, then the subgrid we are partitioning is vacuous and hence there are 0 districts satisfying any claimed conditions.

Otherwise, $\square_{abcd}$ includes elements of V that need to be partitioned. There are two options. One, we may leave $\square_{abcd}$ as one district, but only if it is in $\mathcal{R}$ and $(1-\rho)\frac{\mu(V)}{N} \leq \mu(\square_{abcd}) \leq (1+\rho)\frac{\mu(V)}{N}$. In that case, the number of districts contributing to Player 1's slate is 1 if $\square_{abcd}$ satisfies $\mu_R(\square_{abcd}) \geq 0$ and 0 otherwise (if it is not possible, $\square_{abcd}$ is not a singleton and hence the second option will be possible). Two, we may subdivide $\square_{abcd}$ by cutting it vertically (assuming $a \neq b$) in one of $b-a$ positions or horizontally (assuming $c \neq d$) in one of $d-c$ positions and only further partitioning each of these subdivisions (no districts crossing this cut). In that case, the maximum number of districts contributing to Player 1's slate is the maximum sum (over all positions to make the cut) of the maximum number of such districts for each resultant subgrid, since a hierarchical partition of $\square_{abcd}$ with a horizontal or vertical cut across which there may be no districts is exactly comprised of a hierarchical partition of each subgrid. Finally, the maximum number of districts contributing to Player 1's slate from $\square_{abcd} \backslash V_-$ is the maximum number resulting from each option. Hence if we suppose that COMPUTE-OPT(a', b', c', d') has the desired behavior for all $\square_{a'b'c'd'} \subsetneq \square_{abcd}$, the claim follows by induction on the size of the subgrid $(b-a)(d-c)$ (if it reaches 1, then no recursive calls are made). □

4.3.3 The Redistricting Game

One potential solution for mitigating the stark gerrymandering power described by Theorem 4.2.3 is to use an approach considered by several others in the literature: treat redistricting as a game where both players are self-interested but must take turns choosing districts. More specifically, in [30], the authors propose the I-cut-you-freeze protocol, inspired by protocols such as I-cut-you-choose in the field of fair division [4], in which one player chooses a partition of the (remaining) state into districts and the other player selects one of the districts to appear in the final redistricting, swapping roles after each round. In the streamlined setting where geometry is collapsed (the state is an interval, and non-contiguous districts allow players to cut districts with arbitrary loyalty fractions), each player asymptotically achieves proportional representation when $p_1 = .5$, but with the slate changing quadratically as p_1 deviates from .5 on either side, not linearly. Here, we mention an alternative where instead the players take turns making hierarchical cuts, effectively precluding any districts from crossing a chosen boundary and yielding new redistricting subproblems.

Definition 4.3.10. *The* REDISTRICTING GAME WITH HIERARCHICAL RECTANGLES *is as*

follows: given a tuple $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ as above, Player 1 and Player 2 alternate making hierarchical cuts in V such that after $N-1$ cuts, a valid hierarchical rectangle redistricting is formed.

Notably, a single instance is capable of yielding both the upper bound and lower bound on the gap between Player 1's slate and their slate under proportional representation, with the first cut (player) determining which of these extreme outcomes is realized. Such instances are possible for arbitrarily large N, albeit seemingly more likely for small values of N.

Theorem 4.3.11. *Let $.25 < p_1 \leq .5$. There exists an instance $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ of* Redistricting Game with Hierarchical Rectangles *where $V_- = \emptyset$ and Player 1 has total loyalty fraction p_1 such that, if the first cut is horizontal, Player 1 wins $N_1 = \lfloor Np_1 \rceil$ districts, whereas if the first cut is vertical, Player 1 wins $N_1 = \left\lceil \frac{2(p_1 - .5 - .5\rho)N}{1-\rho} \right\rceil$ districts.*

Proof. Let $V = [1,p] \times [1,p]$ and $N = p$, where $p > 2$ is a prime number such that $p_1 - .25 \geq \frac{1}{4p}$. Set $\rho < \frac{1}{p}$ and $\mu(\{v\}) = \frac{1}{p}$ for all $v \in V$. Let the precinct loyalty fraction for Player 1 be p_+ for each precinct that is in both the $\frac{p+1}{2}$ topmost rows and the $\lfloor 2p_1 p \rfloor$ leftmost columns; otherwise, the precinct loyalty fraction is p_-, where $p_- \in [.25 + \frac{1}{4p}, .5)$ and $p_+ = \frac{2p_1 p^2}{(p+1)\lfloor 2p_1 p \rfloor} - p_- \frac{(p-1)(p - \lfloor 2p_1 p \rfloor)}{(p+1)\lfloor 2p_1 p \rfloor}$. These choices satisfy the equation $p_+ \frac{\frac{p+1}{2}\lfloor 2p_1 p \rfloor}{p^2} + p_- \frac{\frac{p-1}{2}(p - \lfloor 2p_1 p \rfloor)}{p^2} = p_1$.

Observe that the only admissible rectangles in a valid hierarchical rectangle redistricting have exactly p precincts, hence have dimensions $p \times 1$ or $1 \times p$. Such rectangles correspond to the rows and columns of V, respectively. If the first cut is horizontal, the final redistricting is comprised of the rows of V, and by construction Player 1 gets $N_1 = \frac{p+1}{2}$ districts (since $|p_+ - .5| \geq |.5 - p_-|$), which is equal to $\lfloor Np_1 \rceil$ if $p_1 = .5$. If the first cut is vertical, the final redistricting is comprised of the columns of V, and by construction Player 1 gets $N_1 = \lfloor 2p_1 p \rfloor$ districts (since $|p_+ - .5| \geq |.5 - p_-|$), which is equal to $\left\lfloor \frac{2p_1 N}{1-\rho} \right\rfloor$ for ρ sufficiently small. □

By transposing the roles of Player 1 and Player 2, the theorem reveals that, even when proportional representation is attainable via hierarchical rectangles, the very first cut placed may bifurcate the redistricting outcome: either the players both achieve a slate close to proportional representation, or they both experience the largest possible deviation from proportional representation but in opposite directions.

4.4 An Urban-Rural Model

Informed by the worst-case results of Section 4.3, we impose additional structure on the redistricting scenarios we study in the form of a geometric model inspired by real-world

political entities in which an urban center, often in the corner, votes for a particular party, whose support then decreases as a function of the distance from this center. In [5], the authors propose a similar model where (for the purposes of their results) one urban center is chosen, and voters are stochastically converted to the urban party with probability decreasing in their distance from the center. This models random factors beyond geography that shape a person's vote, but assumes it is independent of the decisions made by others in the precinct, and results in a lot of variation in precinct loyalty fractions due to the small population assigned to each (10). Here, we focus more on theoretical analysis and study the case where the decrease is linear in the l_1 distance from the center, also assessing the performance of GERRYMANDERING WITH HIERARCHICAL RECTANGLES with choices of L, W, and N that resemble the actual number of precincts and districts some US states are allocated. While tractability may be one advantage in the l_1 distance assumption, it also reflects the fact that a majority of people access an urban center by car, and most highways (along with city blocks themselves) use a North-South or East-West orientation.

We consider the urban core to be at $(1,1)$ but take $V = [1,L] \times [1,W]$, regarding each ordered pair in V to be the top-right corner of a 1×1 square. Define the following function on $\mathbb{R}^2$

$$p_1(x,y) = \begin{cases} .5 + C\delta - (x+y)\delta, & x+y \leq 2C \\ .5 - C\delta, & x+y > 2C, \end{cases}$$

where $0 < C \leq \frac{L+W}{2}$ represents the distance from the urban core beyond which the population is perfectly split by party (hence the rural landscape is eventually reached), with the loyalty fraction for Player 1 assuming the value of .5, and $\delta \in (0, \frac{1}{2C}]$ is the rate at which Player 1 loses support away from the urban center. A precinct represented by $(b,d) \in V$ has precinct loyalty fraction $\int\int_{[b-1,b]\times[d-1,d]} p(x,y)dxdy$, which simplifies to $p(b-.5, d-.5)$ in the case that $b+d \leq 2C$. We also suppose that each precinct has the same population $\mu := \frac{N}{L^2}$, which attempts to capture the fact that many political entities have legal requirements on the population of each precinct [26] and that, even within the same city, factors such as de facto segregation may result in non-uniform support for the urban party. Therefore, V may be thought of as a cartogram weighted by population more than a traditional 2D map.

The above modeling choices lead to the following definition. In the *urban core* case of GERRYMANDERING WITH RECTANGLES, the problem instance $(V, V_-, E, \mathcal{R}, \mu, \mu_R, \rho, N)$ is such that $V = ([1,L] \times [1,W]) \backslash V_-$, $\mu(V) = N$, $\mu(\{(x,y)\}) = \frac{N}{L^2}$ for all

$(x, y) \in V$, and μ_R induces the above precinct loyalty function: $\mu_R(\{(x, y)\}) = \int\int_{[b-1,b]\times[d-1,d]}[\frac{N}{L^2}p_1(x, y) - \frac{N}{L^2}(1 - p_1(x, y))]dxdy$, where $0 < C \leq \frac{L+W}{2}$ and $\delta \in (0, \frac{1}{2C}]$.

Our objective is to determine whether imposing these constraints that are reminiscent of some real-world dynamics results in any meaningful reduction in the players' ability to gerrymander. To start the analysis, we pass to a continuous relaxation of the model in which the grid is infinitely subdivisible, allowing players to slice individual precincts, so long as this agrees with the population requirements for each district (which in the continuous setting we consider to be exact as if $\rho = 0$). In fact, the motivation for defining population regularity was similar, and that property will be key to converting our results back to the discrete model while only incurring mild quantitative constants.

Lemma 4.4.1. *Let $W \leq L$. In the urban core case of* Gerrymandering with Rectangles *where $V_- = \emptyset$,*

$$p_1 = \begin{cases} 0.5 - C\delta\left(1 - \frac{4}{3}\frac{C^2}{LW}\right), & 0 < C \leq \frac{W}{2} \\ 0.5 - C\delta + \frac{\delta}{6LW}\left[8C^3 - (2C - W)^3\right], & \frac{W}{2} < C \leq \frac{L}{2} \\ 0.5 + C\delta - \frac{L+W}{2}\delta + \frac{4}{3}\frac{\delta}{LW}\left(\frac{L+W}{2} - C\right)^3, & \frac{L}{2} < C \leq \frac{L+W}{2}. \end{cases}$$

Proof. Assume without loss of generality $W \leq L$. We start with the continuous relaxation. To determine the overall population fraction p_1 for Player 1, we integrate the continuous population density function over the rectangular domain $[0, L] \times [0, W]$. The function $p_1(\cdot, \cdot)$ may be rewritten as

$$p_1(x, y) = 0.5 - C\delta + \delta \max(0, 2C - x - y).$$

The total loyalty fraction p_1 is the average of this precinct loyalty fraction on V:

$$p_1 = \frac{1}{LW}\iint_V p_1(x, y)dxdy = 0.5 - C\delta + \frac{\delta}{LW}\iint_V \max(0, 2C - x - y)dxdy,$$

so for $S \subset \mathbb{R}^2$ define

$$I_S = \iint_S \max(0, 2C - x - y)dxdy.$$

Let T be the right isosceles triangle in the first quadrant bounded by $x + y \leq 2C$. Now $I_T = \int_0^{2C}\int_0^{2C-x}(2C - x - y)dydx = \frac{8}{6}C^3 = \frac{4}{3}C^3$.

Because the domain V is bounded by $x = L$ and $y = W$, the geometry of the intersection $V \cap T$ transitions through three distinct phases depending on the magnitude of $2C$. The

inequalities stated for each case are equivalent to the respective inequalities in the piecewise expression for p_1 since $\min(L, W) = W$ throughout the proof.

Case 1: The Origin Triangle ($0 < 2C \leq W$). The triangle T is fully contained within V. Therefore, $I_V = I_T = \frac{4}{3}C^3$. Substituting this into the population equation yields:

$$p_1 = 0.5 - C\delta + \frac{\delta}{LW}\left(\frac{4}{3}C^3\right) = 0.5 - C\delta + \frac{4}{3}\delta C\frac{C^2}{LW} = 0.5 - C\delta\left(1 - \frac{4}{3}\frac{C^2}{LW}\right).$$

Case 2: The Trapezoidal Phase ($W < 2C \leq L$). The region T crosses the boundary $y = W$ but remains bounded by $x = L$. The volume outside V is a smaller tetrahedron situated above $y = W$. By applying the coordinate shift $y' = y - W$, the volume of this protruding region T_W is identical in form to I_T but with legs of length $(2C - W)$. Thus, $I_{T_W} = \frac{1}{6}(2C - W)^3$. To rigorously calculate I_{T_W}, the volume of the region where the density function exceeds the top boundary of the rectangular domain ($y = W$), we set up the double integral over the protruding triangle T_W. The region T_W is defined by the inequalities $x \geq 0$, $y \geq W$, and $x + y \leq 2C$. Therefore, the bounds of integration are y from W to $2C$, and x from 0 to $2C - y$. The integral is:

$$I_{T_W} = \int_W^{2C}\int_0^{2C-y}(2C - x - y)dxdy.$$

First, we evaluate the inner integral with respect to x:

$$\int_0^{2C-y}(2C - y - x)dx = \left[(2C - y)x - \frac{1}{2}x^2\right]_0^{2C-y}.$$

Substituting the upper bound $x = 2C - y$, this simplifies to

$$(2C - y)(2C - y) - \frac{1}{2}(2C - y)^2 = (2C - y)^2 - \frac{1}{2}(2C - y)^2 = \frac{1}{2}(2C - y)^2.$$

Next, we evaluate the outer integral with respect to y:

$$I_{T_W} = \int_W^{2C}\frac{1}{2}(2C - y)^2dy = \int_0^{2C-W}\frac{1}{2}u^2du.$$

Evaluating this yields:

$$I_{T_W} = \left[\frac{1}{6}u^3\right]_0^{2C-W} = \frac{1}{6}(2C - W)^3,$$

so write

$$I_V = I_T - I_{T_W} = \frac{8C^3}{6} - \frac{(2C - W)^3}{6}.$$

Using the above form of p_1 and replacing W by $\min(L, W)$ gives:

$$p_1 = 0.5 - C\delta + \frac{\delta}{6LW}\left[8C^3 - (2C - \min(L, W))^3\right].$$

Case 3: The Opposite Corner Triangle ($L < 2C \leq L + W$). The region T now crosses both boundaries $y = W$ and $x = L$. Because $2C \leq L + W$, T does not cover the opposite corner (L, W), meaning the regions protruding beyond $y = W$ and $x = L$ are disjoint. We subtract both protruding volumes: $I_{T_W} = \frac{1}{6}(2C - W)^3$ and $I_{T_L} = \frac{1}{6}(2C - L)^3$. Now

$$I_V = I_V - I_{T_W} - I_{T_L} = \frac{1}{6}\left[8C^3 - (2C - W)^3 - (2C - L)^3\right].$$

Expanding the polynomials yields

$$6I_V = -8C^3 + 12C^2(L + W) - 6C(L^2 + W^2) + L^3 + W^3.$$

To reveal the symmetry with Case 1, we rewrite this expression by completing the cube for the term $\left(\frac{L+W}{2} - C\right)$. Note that

$$8\left(\frac{L + W}{2} - C\right)^3 = (L + W)^3 - 6(L + W)^2C + 12(L + W)C^2 - 8C^3.$$

Subtracting this perfect cube from our expanded expression leaves a remainder of $12CLW - 3LW(L + W)$. Thus

$$6I_V = 12CLW - 3LW(L + W) + 8\left(\frac{L + W}{2} - C\right)^3.$$

Dividing by 6 gives

$$I_V = 2CLW - LW\left(\frac{L + W}{2}\right) + \frac{4}{3}\left(\frac{L + W}{2} - C\right)^3.$$

Finally, substituting I_V back into our equation for p_1:

$$p_1 = 0.5 + C\delta - \delta\left(\frac{L + W}{2}\right) + \frac{4\delta}{3LW}\left(\frac{L + W}{2} - C\right)^3.$$

Note that in the discrete setting, $p_1 = \frac{1}{LW}\sum_{y=1}^{W}\sum_{x=1}^{L} p(x, y) = \frac{1}{LW}\int\int_V p_1(x, y)dxdy$, so the piecewise expression applies to the original urban core case without continuous relaxation. □

Lemma 4.4.2. *Assume $W \leq L$. The urban core case of* Gerrymandering with Rect-

ANGLES *where* $V_- = \emptyset$ *holds satisfies that for all contiguous redistrictings,* $N_1 \leq Fp_1 N$*, where*

$$Fp_1 = \begin{cases} \frac{1}{1-\rho}\frac{9C^2}{8LW}, & 0 < C \leq \frac{2}{3}W \\ \frac{1}{1-\rho}\frac{C-\frac{W}{2}+\sqrt{C^2-CW+\frac{W^2}{3}}}{L}, & \frac{2}{3}W < C \leq \frac{3L^2-W^2}{6L-3W} \\ \frac{1}{1-\rho}(1-\frac{h^2}{2LW}), & \frac{3L^2-W^2}{6L-3W} < C \leq \frac{L+W}{2}, \end{cases}$$

where h *is a root in* $[0, W]$ *to the cubic equation* $2h^3-3(L+W-C)h^2+3LW(L+W-2C) = 0$*, which exists uniquely.*

Proof. Assume without loss of generality $W \leq L$. We first consider the continuous relaxation. Fix a contiguous redistricting and let U be the union of all districts in Player 1's slate. Therefore, $\iint_U p_1(x, y)dxdy \geq .5\text{Area}(U)$, which is equivalent to

$$\iint_{[0,L]\times[0,W]} f(x, y)g(x, y)dxdy \leq 0,$$

where $f(x, y) = .5 - p_1(x, y)$ and $g(x, y) = \mathbf{1}_{\{(x,y)\in U\}}$. By the bathtub principle from real analysis, if $\text{Area}(U)$ is fixed, the integral on the left-hand side is minimized by setting U to consist of all points for which f is smaller than some threshold value, i.e. $p_1(x, y)$ is greater than some threshold value. Observe that for sufficiently large values, the level sets for $p_1(x, y)$ are lines of the form $x + y = c$ for c arbitrarily small. The corresponding optimal region U is thus defined by a set of the form $\{(x, y) \in [0, L] \times [0, W] : x + y \leq K\}$, provided $K \leq 2C$. Because $f(x, y)$ is linear in this domain, minimizing the integral is equivalent to setting the coordinate sum of the centroid of U equal to C (i.e., $x_c + y_c = C$). We evaluate this centroid across three geometric phases of V.

Case 1: The Origin Triangle $(0 < K \leq W)$. The region U is a right isosceles triangle with legs K. Its centroid coordinate sum is $\frac{2K}{3}$. Setting $\frac{2K}{3} = C$ yields $K = 1.5C \leq 2C$. The claimed triangular shape of U is valid for $0 < K \leq W$, i.e., $C \leq \frac{2}{3}W$. The area of U is $\frac{1}{2}(1.5C)^2 = \frac{9}{8}C^2$. The area fraction is $p_1F = \frac{9C^2}{8LW}$.

Case 2: The Trapezoidal Phase $(W < K \leq L)$. The region U is a right trapezoid bounded by $y = W$. We decompose it into a rectangle of size $(K - W) \times W$ and a right triangle of legs W. Taking the area-weighted average of their centroids gives the coordinate sum of the trapezoid: $x_c + y_c = \frac{W(K-W)(\frac{K}{2})+\frac{W^2}{2}(K-\frac{W}{3})}{W(K-W)+\frac{W^2}{2}} = \frac{3K^2-W^2}{3(2K-W)}$. Setting this equal to C yields the quadratic equation $3K^2 - 6CK + (3CW - W^2) = 0$, which has the root $K = C + \sqrt{C^2 - CW + \frac{W^2}{3}}$, which is greater than W because $C > \frac{2}{3}W$ (for the same reason, the other root is less than W as then $\sqrt{C^2 - CW + \frac{W^2}{3}}$ is positive because the leading

coefficient of the quadratic therein is positive whereas its discriminant is negative, meaning it has no real roots). Likewise, the root is at most $2C$. The claimed trapezoidal shape of U is valid for $W < K \leq L$, i.e., $\frac{2}{3}W < C \leq \frac{3L^2-W^2}{6L-3W}$. The area of U is $WK - \frac{W^2}{2}$. Dividing by LW yields the area fraction $p_1F = \frac{K-\frac{W}{2}}{L}$.

Case 3: The Pentagon Phase ($L < K \leq L+W$). The region U is a pentagon equal to to the full rectangle LW minus a right isosceles triangle of leg length $h = L+W-K$ at the top-right corner. As in Case 2, we can calculate the centroid sum $x_c+y_c = \frac{LW(\frac{L+W}{2})-\frac{1}{2}h^2(L+W-\frac{2h}{3})}{LW-\frac{1}{2}h^2}$. Setting this sum to C and rearranging, $2h^3 - 3(L+W-C)h^2 + 3LW(L+W-2C) = 0$, where we take h to be a solution to the equation and in the interval $[0, W]$. To verify that h is well-defined, let $P(h) = 2h^3 - 3(L+W-C)h^2 + 3LW(L+W-2C)$. The derivative is $P'(h) = 6h(h-(L+W-C))$. Under the Case 3 assumption $C \leq \frac{L+W}{2}$ and the assumption $W \leq L$, the first critical point is 0 and the second critical point is $L+W-C \geq W$. By inspection, $P'(h) \leq 0$ for all $h \in [0, W] \subset [0, L+W-C]$, making $P(h)$ strictly decreasing on the former. Further, $P(0) = 3LW(L+W-2C) \geq 0$, and $P(W) = W(3L^2-W^2)-3CW(2L-W) < 0$ because $C > \frac{3L^2-W^2}{6L-3W}$. Thus by the intermediate value theorem and strict monotonicity, there exists exactly one real root $h \in [0, W]$. Because $h = L+W-K$, the necessary condition $K \leq 2C$ is mathematically equivalent to requiring $h \geq L+W-2C$. However, $P(L+W-2C) \geq 0$. Therefore, the unique root h must be greater than or equal to $L+W-2C$, which establishes $K \leq 2C$. The claimed pentagonal shape of U is valid for $L < K \leq L+W$, i.e., $\frac{3L^2-W^2}{6L-3W} < C \leq \frac{L+W}{2}$ (the maximum coordinate sum of the domain is $L+W$). The area of U is $LW-\frac{1}{2}h^2$, yielding the area fraction $p_1F = 1-\frac{h^2}{2LW}$.

Note that under our cost function, each square in V is equivalent to taking a single 1×1 region into U, so the above areas of U are an upper bound in the original urban core case as well. Moreover, each district can have area as small as $1-\rho$, so we divide by that quantity. □

On the other hand, Player 1 can get reasonably close to this upper bound (depending on the parameter values) by adopting a "bookcase" hierarchical partition.

Theorem 4.4.3. *Suppose $W \leq L$. In the urban core case of* Gerrymandering with Rectangles *where $V_- = \emptyset$ and (PR) holds, there exists a valid hierarchical redistricting such that $N_1 \geq Ap_1N - .25\ln\left(\frac{L}{W}N\right) - .75$, where*

$$Ap_1 = \max\left(\frac{C}{Wk_L} - \frac{L}{2Wk_L^2} + \frac{L}{4W(k_L+1)^2}, \frac{C}{Lk_W} - \frac{W}{2Lk_W^2} + \frac{W}{4L(k_W+1)^2}\right),$$

$k_L = \left\lceil \frac{L}{2C} \right\rceil$, *and* $k_W = \left\lceil \frac{W}{2C} \right\rceil$.

Proof. Assume without loss of generality that $W \leq L$. Let Player 1 construct a sequence of horizontal segments (subsets of contiguous rows) from bottom to top along the y-axis, with each segment being subsequently divided vertically into districts of the same height. Any district in a segment partitioned into k districts (called a segment of type k) has length $l_k = \frac{L}{k}$ and width $w_k = \frac{LW/N}{l_k} = \frac{kW}{N}$.

To ensure the leftmost district in each segment adds to Player 1's slate, we require that its centroid (x_c, y_c) satisfy $x_c + y_c \leq C$. Substituting the physical dimensions of the centroid for a segment whose bottom-most point has vertical coordinate y ("bottom anchor point"), yields $l_k/2 + y + w_k/2 \leq C$. This allows us to define an upper bound B_k for the bottom anchor point y of a segment of type k as follows:

$$B_k = C - \left(\frac{L}{2k} + \frac{kW}{2N} \right).$$

The quantity $\left(\frac{L}{2k} + \frac{kW}{2N}\right)$, $k > 0$ attains its global minimum at $k = \sqrt{\frac{L}{W}N}$. Therefore, $B_{k-1} \leq B_k$ for all $k \leq \left\lfloor \sqrt{\frac{L}{W}N} \right\rfloor =: K_{\max}$.

Player 1 begins packing horizontal segments at the origin ($y = 0$) flush against each other. Setting $k_L = \max\left(\left\lceil \frac{L}{2C} \right\rceil, 0\right)$, Player 1 packs identical segments of type k_L until the bottom anchor point of the next segment would be strictly greater than B_{k_L}. Player 1 then increments the segment type to k_L+1, repeating this process for each boundary B_k until $K_{\max}$.

Let y_k denote the actual starting y-coordinate for the first segment of type k, with $y_{k_L} = 0$. By the defined stopping condition, the start of the next block of segments must satisfy $y_{k+1} > B_k$. The exact integer number of districts contributing to Player 1's slate, N_1, is the sum of the actual segments packed in each phase:

$$N_1 = \sum_{k=k_L}^{K_{\max}} \frac{y_{k+1} - y_k}{w_k},$$

where $y_{K_{\max}+1}$ is defined as the ending y-coordinate for the last segment of type $K_{\max}$. We compare this to the non-integral approximation

$$S_{\text{cont}} = \sum_{k=k_L}^{K_{\max}} \frac{B_k - B_{k-1}}{w_k},$$

where $B_{k_L-1} := 0$. By applying summation by parts to both series, we rearrange the series as follows:

$$N_1 = \sum_{k=k_L}^{K_{\max}-1} y_{k+1}\left(\frac{1}{w_k} - \frac{1}{w_{k+1}}\right) + \frac{y_{K_{\max}+1}}{w_{K_{\max}}}$$

and

$$S_{\text{cont}} = \sum_{k=k_L}^{K_{\max}-1} B_k\left(\frac{1}{w_k} - \frac{1}{w_{k+1}}\right) + \frac{B_{K_{\max}}}{w_{K_{\max}}}.$$

Because $w_{k+1} > w_k$, the weighting term $\left(\frac{1}{w_k} - \frac{1}{w_{k+1}}\right)$ is strictly positive. Because $y_{k+1} > B_k$ for all k, every term in the former sum exceeds the corresponding term in the latter sum. Therefore, $N_1 > S_{\text{cont}}$.

Substituting our definitions into S_{cont} yields the strict lower bound:

$$N_1 > N\left(\frac{C}{Wk_L} - \frac{L}{2Wk_L^2}\right) - .5 + \sum_{k=k_L+1}^{K_{\max}}\left(\frac{NL}{2Wk^2(k-1)} - \frac{1}{2k}\right).$$

Because $\frac{1}{2k^2(k-1)} \geq \frac{1}{2k^3}$ for all $k \geq 2$, we have

$$\sum_{k=k_L+1}^{K_{\max}} \frac{NL}{2Wk^2(k-1)} \geq \frac{NL}{W}\int_{k_L+1}^{K_{\max}+1} \frac{1}{2t^3}dt = \frac{NL}{4W}\left(\frac{1}{(k_L+1)^2} - \frac{1}{\frac{NL}{W}}\right) = \frac{NL}{4W(k_L+1)^2} - .25,$$

so

$$\sum_{k=k_L+1}^{K_{\max}} \frac{NL}{2Wk^2(k-1)} > \frac{NL}{4W(k_L+1)^2} - 0.25.$$

Bounding the harmonic partial sum, $\sum_{k=k_L+1}^{K_{\max}} \frac{1}{2k} < .5\ln\left(\sqrt{\frac{L}{W}N}\right) = .25\ln\left(\frac{L}{W}N\right)$.

Combining inequalities, we get:

$$N_1 > N\left(\frac{C}{Wk_L} - \frac{L}{2Wk_L^2} + \frac{L}{4W(k_L+1)^2}\right) - .25\ln\left(\frac{L}{W}N\right) - 0.75.$$

To convert back to the actual urban core case, we elect to partition the remaining portion of V that is above the horizontal segments of type $K_{\max}$ using horizontal segments of type 1. Therefore, the entire grid is partitioned using horizontal segments. Recall that each row has the same population in the urban core case. For $k \in \mathbb{N}$, we define $a_k = k - \left\lfloor \frac{k}{\mu(V_{i,\cdot})} \right\rfloor \mu(V_{i,\cdot})$ to

be the distance between k times the mean district population and the greatest population a subset of contiguous rows can achieve that is at most k times the mean district population, while $b_k = \lceil \frac{k}{\mu(V_{i,\cdot})} \rceil \mu(V_{i,\cdot}) - k$ is the distance between k times the mean district population and the smallest population a subset of contiguous rows can achieve that is at least k times the mean district population. Observe that $a_k + b_k = \mu(V_{i,\cdot})$, unless $\frac{k}{\mu(V_{i,\cdot})} \in \mathbb{N}$. We place the same number of horizontal segments of each type and in the same order as for the continuous relaxation. Let S_m be the population not in the bottommost m segments in our discrete construction minus that same quantity in the continuous relaxation, where $m \geq 0$ and $S_0 := 0$. Suppose that the $(m+1)$st bottommost horizontal segment is of some type k in the continuous relaxation. Suppose this is not the topmost horizontal segment in the continuous relaxation. If $\frac{k}{\mu(V_{i,\cdot})} \in \mathbb{N}$, then in our discrete construction this horizontal segment is formed using a_k rows. Otherwise, in our discrete construction, this horizontal segment is formed using a_k rows if $S_m \leq b_k$ and b_k rows if $S_m > b_k$. In the first case, $S_{m+1} \leq a_k + b_k = \mu(V_{i,\cdot})$, and in the second case, $S_{m+1} \geq b_k - b_k = 0$. Thus $S_m \in [0, \mu(V_{i,\cdot})]$ for all m (indicating if anything a downward shift compared to the positioning in the continuous relaxation), and by construction our horizontal segment has population in $[k-\rho, k+\rho]$, so by Lemma 4.3.6, we may use vertical cuts to partition it into k districts, where by symmetry the leftmost district may be taken to use at most $\frac{L}{k}$ rows. If instead this is the topmost horizontal segment, we still know (from analyzing the preceding segments as above) that $S_m \in [0, \mu(V_{i,\cdot})]$, so the remaining population for the segment in the discrete setting is in $[k, k+\rho]$ (indicating if anything a downward shift compared to the positioning in the continuous relaxation), and it may be vertically partitioned as in the preceding sentence. Synthesizing these facts, in each district contributing to Player 1's slate in the above analysis, the centroid may only move down and to the left in our discrete construction, hence still having precinct loyalty fraction for Player 1 at least .5. Thus our lower bound applies to the original urban core case without continuous relaxation.

Analogously, we may invert the role of the axes by transposing L and W in all of our expressions, with the claimed bound obtained by taking the maximum lower bound from these two arguments. □

We now analyze a few structured cases with Lemma 4.4.2 and Theorem 4.4.3, with particular interest in the asymptotic proportion of districts we can guarantee for Player 1 compared to the theoretical limit, namely the geometric efficiency $\frac{A}{F}$, as well as the total loyalty fraction p_1.

Corollary 4.4.4. *In the urban core case of* Gerrymandering with Rectangles *where*

$V_- = \emptyset$ and (PR) holds, suppose $W \leq L$ and $0 < C \leq \frac{2}{3}W$. As $\frac{L}{C} \to \infty$, $A \to \frac{C^2}{LW}$ and $\frac{A}{F} \to (1-\rho)\frac{8}{9}$.

Proof. Because $C \leq \frac{2}{3}W$, we are in Case 1 of Lemma 4.4.2, so $F = \frac{1}{1-\rho}\frac{1}{p_1}\frac{9C^2}{8LW}$. On the other hand, because $\frac{L}{C} \to \infty$, the effect of rounding to obtain the first segment type in Theorem 4.4.3 is negligible; $\frac{k_L}{\frac{L}{2C}} \to 1$, $\frac{k_W}{\frac{W}{2C}} \to 1$, $\frac{(k_L+1)^2}{k_L^2} \to 1$, and $\frac{(k_W+1)^2}{k_W^2} \to 1$. Making the substitutions suggested by these limits, $A \to \frac{1}{p_1}(\frac{2C^2}{LW} - \frac{4C^2}{2LW} + \frac{C^2}{LW}) = \frac{1}{p_1}\frac{C^2}{LW}$. Therefore, $\frac{A}{F} \to (1-\rho)\frac{8}{9}$. □

Unfortunately, when $C \leq \frac{2}{3}W$ and $\frac{L}{C} \to \infty$, we have that the region $x + y \leq C$ is very small, and by symmetry p_1 is close to $.5 - C\delta$. This renders the scenario less realistic, especially in the sense that Player 1 would be unlikely to control gerrymandering in such a state.

It is possible, however, to have $p_1 \approx .5$ and $A \approx 1$, meaning that constraining Player 1 to hierarchical rectangles at least gives them the amount that would be guaranteed under proportional representation.

Corollary 4.4.5. *In the urban core case of* Gerrymandering with Rectangles *where $V_- = \emptyset$ and (PR) holds, for $p_1 \in [.4375, .5]$, there exists an instance in which $A \in [.875, 1]$ and $F \in [\frac{1}{1-\rho}1.662, \frac{1}{1-\rho}1.897]$.*

Proof. Let $L = W$, $C = \frac{7}{8}W$, and $\delta = \frac{24}{47W}$. By Case 3 of Lemma 4.4.1 (the lower bound for this regime is plainly satisfied), $p_1 = .5 - \frac{1}{8}W\delta + \frac{4}{3}\frac{\delta}{W^2}(\frac{1}{8}W)^3 = .5 - \frac{47}{384}W\delta = .4375$.

Using Case 3 of Lemma 4.4.2 (the lower bound for this regime simplifies to $\frac{2}{3}W$ because there is no asymmetry between the two dimensions in this example), we need the solution to $2h^3 - 3(2W - \frac{7}{8}W)h^2 + 3W^2(2W - \frac{7}{4}W) = 0$, which is equivalent to $16(\frac{h}{W})^3 - 27(\frac{h}{W})^2 + 6 = 0$. This equation admits the solution $h \approx .58255W$, implying that $F \approx \frac{1}{1-\rho}\frac{1}{.4375}(1 - \frac{.58255^2}{2}) \approx \frac{1}{1-\rho}1.898$.

In Theorem 4.4.3, $k_L = k_W = \left\lceil \frac{W}{2(\frac{7}{8}W)} \right\rceil = 1$. Now $\frac{C}{Wk_L} - \frac{L}{2Wk_L^2} + \frac{L}{4W(k_L+1)^2} = \frac{C}{Lk_W} - \frac{W}{2Lk_W^2} + \frac{W}{4L(k_W+1)^2} = \frac{7/8}{1} - \frac{1}{2} + \frac{1}{16} = .4375$, so $A = 1$.

Lastly, we note that by taking $\delta \to 0$ in the above framework, we achieve $p_1 \to .5$ and have that $A \to .875$ (since the product Ap_1 is unchanged) while $F \approx \frac{1}{1-\rho}1.661$ (since the product Fp_1 is unchanged). □

Furthermore, in the case that $L = W$, experimental results indicate that the asymptotic lower bound for hierarchical rectangle redistricting represented by A in the setting of Corollary 4.4.5 is reasonably sharp (regardless of whether (PR) holds), with Figures 4.3, 4.4, 4.5,

and 4.6 illustrating the optimality of constructions reminiscent of the construction in Theorem 4.4.3. Increasing the aspect ratio $\frac{L}{W}$ away from 1 (without changing the proportionality between W and C) tends to decrease the lower bound A, the maximum fraction of proportional representation achievable in Player 1's slate, $\frac{N_1}{Np_1}$, and the upper bound F, though increasing C keeps $A = 1$ achievable. These results are summarized in Table 4.1.

L	W	N	C	δ	ρ	p_1	A	$N_1/(Np_1)$	F
50	50	10	43.75	0.01021	0.01	0.43750	1.00000	1.14286	1.91717
50	50	25	43.75	0.01021	0.01	0.43750	1.00000	1.18857	1.91717
50	50	50	43.75	0.01021	0.01	0.43750	1.00000	1.23429	1.91717
100	100	10	87.50	0.00511	0.01	0.43750	1.00000	1.14286	1.91717
100	100	25	87.50	0.00511	0.01	0.43750	1.00000	1.28000	1.91717
100	100	50	87.50	0.00511	0.01	0.43750	1.00000	1.28000	1.91717
50	50	10	43.75	0.00102	0.01	0.49735	0.87966	1.00533	1.68641
50	50	25	43.75	0.00102	0.01	0.49735	0.87966	1.04554	1.68641
50	50	50	43.75	0.00102	0.01	0.49735	0.87966	1.08575	1.68641
100	25	10	21.875	0.00204	0.01	0.46582	0.28325	0.21467	0.45983
100	25	25	21.875	0.00204	0.01	0.46582	0.28325	0.25761	0.45983
100	25	50	21.875	0.00204	0.01	0.46582	0.28325	0.25761	0.45983
250	10	10	8.75	0.00511	0.01	0.45700	0.05950	0.00000	0.07500
250	10	25	8.75	0.00511	0.01	0.45700	0.05950	0.00000	0.07500
250	10	50	8.75	0.00511	0.01	0.45700	0.05950	0.04376	0.07500
100	100	10	87.50	0.00051	0.01	0.49735	0.87966	1.00533	1.68641
100	100	25	87.50	0.00051	0.01	0.49735	0.87966	1.12597	1.68641
100	100	50	87.50	0.00051	0.01	0.49735	0.87966	1.12597	1.68641
125	80	10	70.00	0.00064	0.01	0.48218	0.58070	0.82957	1.13726
125	80	25	70.00	0.00064	0.01	0.48218	0.58070	0.82957	1.13726
125	80	50	70.00	0.00064	0.01	0.48218	0.58070	0.82957	1.13726
200	50	10	43.75	0.00102	0.01	0.46582	0.28325	0.21467	0.45983
200	50	25	43.75	0.00102	0.01	0.46582	0.28325	0.34348	0.45983
200	50	50	43.75	0.00102	0.01	0.46582	0.28325	0.34348	0.45983

Table 4.1: Parameters and calculated bounds for varying grid dimensions and district counts.

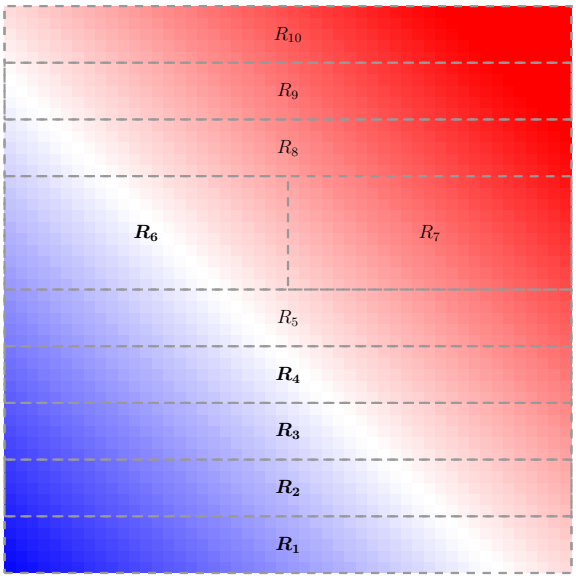


Figure 4.3: A solution to an instance of Gerrymandering with Hierarchical Rectangles, with the resulting districts denoted as R_i (districts won by Player 1 are written in bold). Precincts are colored according to the value of p_1 at them, with values greater than .5 blue, values less than .5 red, and values equal to .5 white). $V_- = \emptyset$. Player 1's slate is $N_1 = 5$. Parameters used: $L = 50, W = 50, N = 10, C = \frac{7}{8}W, \delta = \frac{24}{470W}, \rho = .01$.

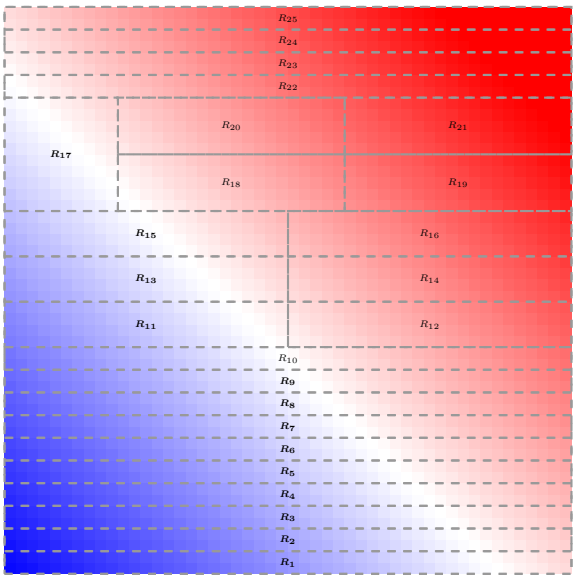


Figure 4.4: A solution to an instance of Gerrymandering with Hierarchical Rectangles, with the resulting districts denoted as R_i (districts won by Player 1 are written in bold). Precincts are colored according to the value of p_1 at them, with values greater than .5 blue, values less than .5 red, and values equal to .5 white). $V_- = \emptyset$. Player 1's slate is $N_1 = 13$. Parameters used: $L = 50, W = 50, N = 25, C = \frac{7}{8}W, \delta = \frac{24}{470W}, \rho = .01$.

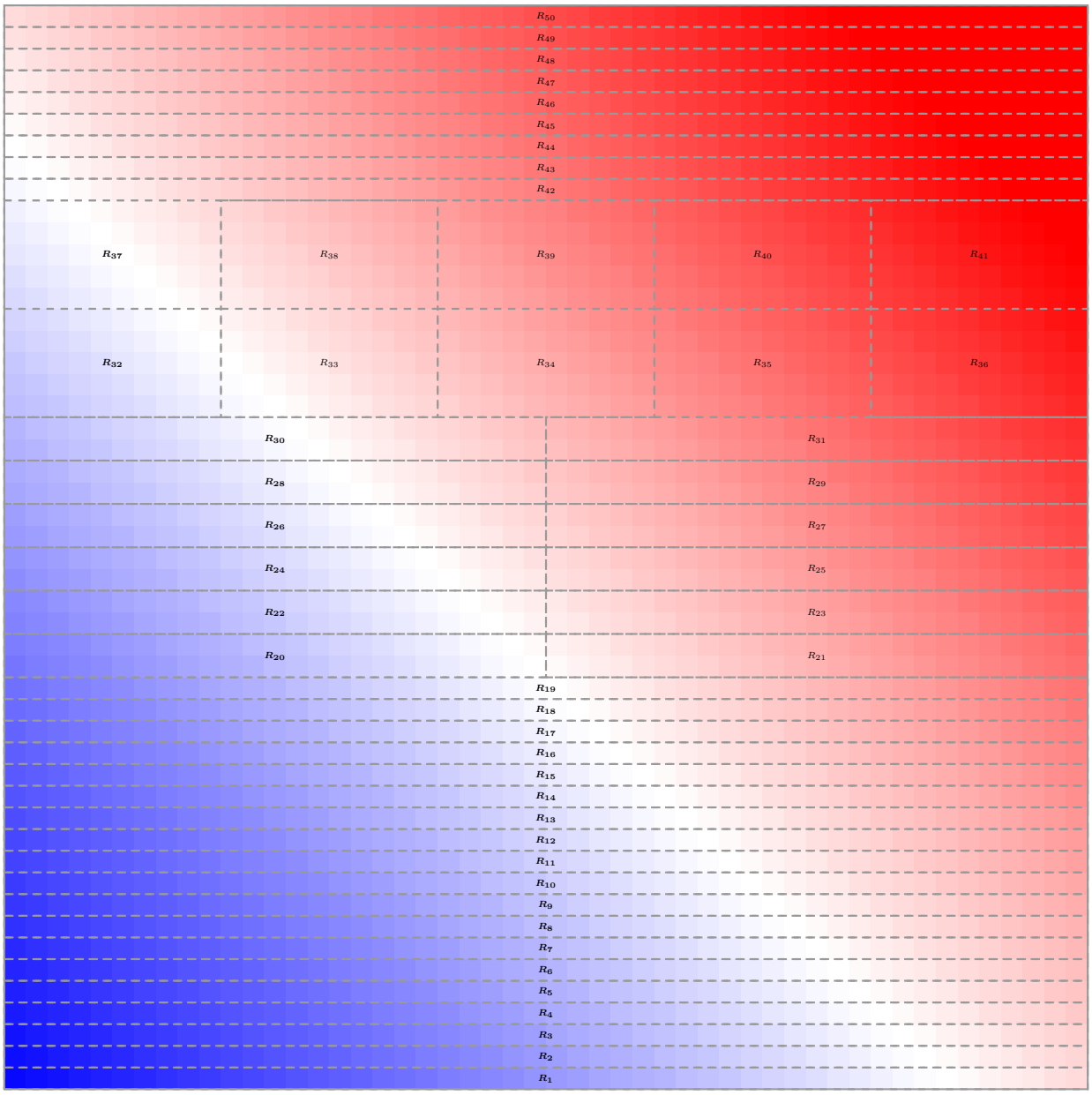


Figure 4.5: A solution to an instance of GERRYMANDERING WITH HIERARCHICAL RECTANGLES, with the resulting districts denoted as R_i (districts won by Player 1 are written in bold). Precincts are colored according to the value of p_1 at them, with values greater than .5 blue, values less than .5 red, and values equal to .5 white). $V_- = \emptyset$. Player 1's slate is $N_1 = 27$. Parameters used: $L = 50, W = 50, N = 50, C = \frac{7}{8}W, \delta = \frac{24}{470W}, \rho = .01$.

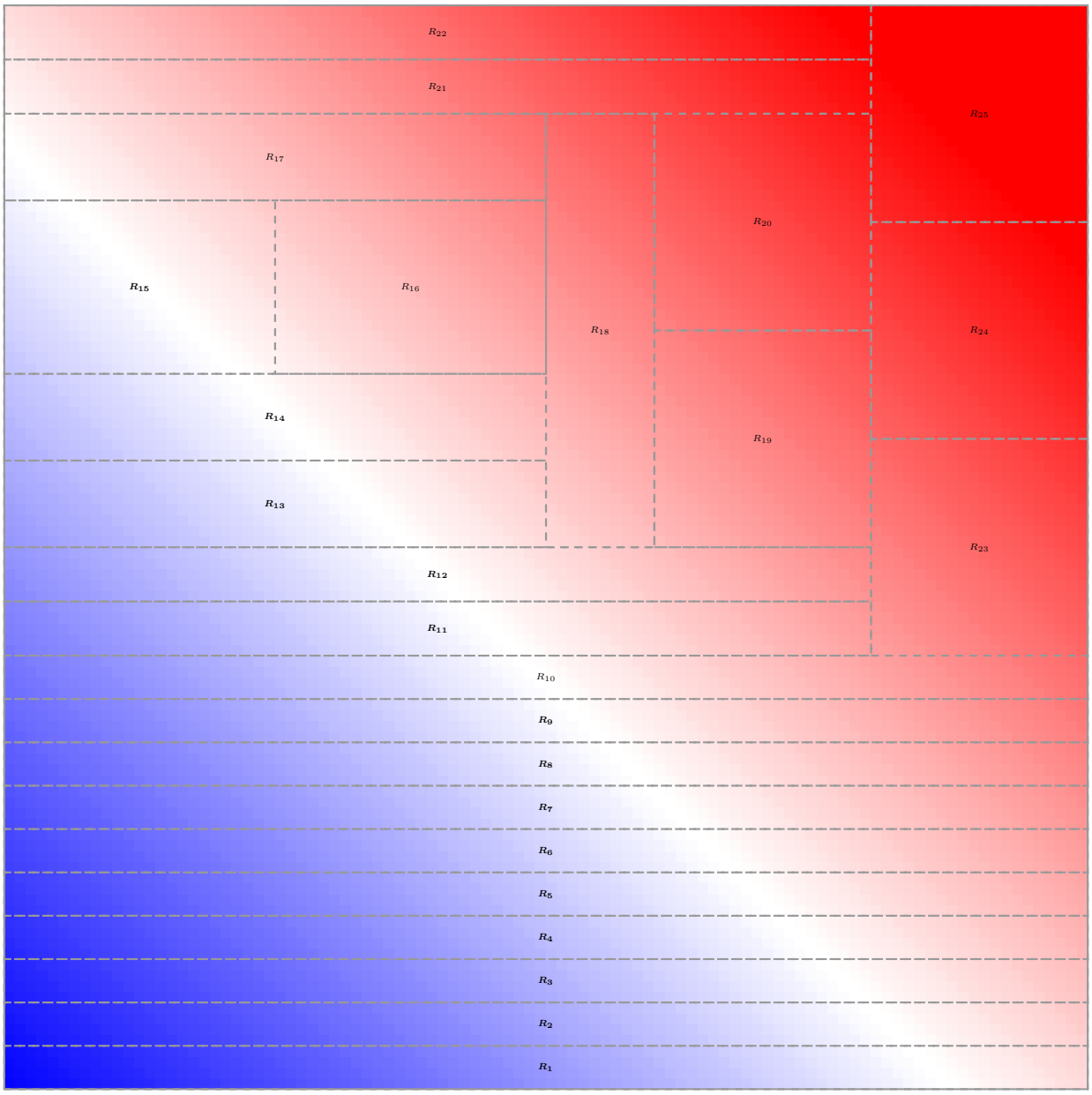


Figure 4.6: A solution to an instance of GERRYMANDERING WITH HIERARCHICAL RECTANGLES, with the resulting districts denoted as R_i (districts won by Player 1 are written in bold). Precincts are colored according to the value of p_1 at them, with values greater than .5 blue, values less than .5 red, and values equal to .5 white. $V_- = \emptyset$. Player 1's slate is $N_1 = 14$. Parameters used: $L = 100, W = 100, N = 25, C = \frac{7}{8}W, \delta = \frac{24}{470W}, \rho = .01$.

CHAPTER 5

Conclusion

In this thesis, we saw several applications of graphs to combinatorial optimization, including a surprisingly general ranking problem that is adaptable to the study of redistricting.
Influence maximization has been proposed through numerous variants in the literature, including negative activation. However, dichotomizing the packets the algorithm designer has at their disposal tends to add intricacies to both the unconstrained and the constrained version of the problem. The former is no longer a matter of inundating the network with information, instead becoming a careful balancing act that has qualitative elements—the structure of the graph's strongly connected components—and quantitative elements—the number of packets of each kind required. The latter is still hard, but may sometimes be understood in terms of simple decision rules about whether to seed packets.

Ranking with partitioning, on the other hand, does not appear to have been studied as standalone problems before this thesis, but addresses many real-world scenarios where objects have been ranked without much attention to the underlying partition implied. While only Rank Minimization is tractable in general (and with a procedure that necessarily results in a very short list under mild conditions), many special cases and variants representing genuine applications make most or even all four problems polynomial-time solvable. The explicit minimum rank and percentile expressions obtained for the complete graph case imply that these quantities are easily manipulated, which is confirmed by the high-stakes numerical experiment for quantifying greenhouse gas emissions, with controversial sources of emissions having a flexible position on ranking lists. Publishing the ranking lists achieving such extremes—or even depicting it as a tree, as is possible for this hierarchical category variant of the problems—illuminates a common thread among such outcomes: aggregating into categories of different scales throughout the similarity-depicting graph.

Redistricting with hierarchical rectangles arises from just a bit of added structure to the

grid with (hierarchical) rectangles variant of the ranking problems: the requirement that each subset have roughly the same size according to the defined measure, interpreted solely as population in this chapter. Compared with the more familiar contiguous redistricting process that underlies the United States House of Representatives, rectangles and hierarchical rectangles constitute two forms of regularization, impeding the expressive power of parties drawing the district lines to encircle specific pockets of their constituencies in a way that disenfranchises others but also in a way that achieves proportional representation. In recognition of this duality, we first look at examples where even well-intentioned parties may not attain proportional representation via (hierarchical) rectangles, but with the next less restrictive model of district. On the other hand, we present an urban-rural model that yields a non-trivial lower bound on a player's redistricting outcome with hierarchical rectangles. Our analogue with the hierarchical rectangles variant of Rank Maximization gives an algorithm for this revised setting, which does not eclipse the state of the art in gerrymandering algorithms, but is executed to confirm that in such a closely divided urban-rural state, gerrymandering ability can be curtailed with hierarchical rectangles so that it merely coincides with proportional representation.

Below, we continue discussing each chapter and identify future directions for each of them.

Chapter 2: Influence Maximization with Signed Information

There are several directions for further work. Most concretely, the results for l_2^2 may be extendable to the objective functions l_1 and l_μ; not much is known about optimization with respect to l_μ, but it captures the change in concavity indicative of diminishing returns to additional information, and attaching probabilities to beliefs leaves open the possibility of considering the expected utility of participants and how that might be maximized or manipulated (as in Bayesian persuasion). Of interest is how robust the results are to the way that bias is measured, albeit in a sign- and order-preserving way.

On a theoretical level, there is a lot unresolved regarding the dependence of Info Bal on the probability of transmission parameters, especially whether an argument can establish the existence of a supercritical regime, and what occurs for intermediate values closer to $\frac{1}{2}$ (beyond NP-hardness). Since each packet induces an Erdős–Rényi random subgraph of the actual social network reflecting whether it will be transmitted along each edge, a key point of interest is whether analogous probability thresholds exist for understanding the transmission of packets or even the optimal solution in the stochastic case.

Unlike preexisting works in influence maximization, submodularity is not a concept that may immediately be applied to Const Info Bal or Info Bal, even though we may regard the set of choices by the algorithm designer as finite due to limits on the number of packets used: either explicitly or due to variance terms, respectively. The most natural approach—creating a finite number (based on the aforementioned limit) of elements for $+1$ and -1 packets for each node representing a seed of that type and interpreting subsets of elements as a set of seeds for the instance—does not result in any of the objective functions being a submodular of this subset. Because the packets are not uniformly beneficial, sometimes things get worse before they get better. For instance, consider an instance with one strongly connected component comprised of two nodes with bias 0. Then seeding the first node a packet of 1 reduces the bias after seeding a packet of -1 but increases the bias if done beforehand, so submodularity does not hold; clearly this conclusion extends to the stochastic case by continuity.

Lemma 2.3.1 suggests that the number of packets prescribed by Solution (2.4) has exponent given by the number of "layers" in the graph (in that example, a subset of vertices that is permuted under an isomorphism of the underlying graph) and base given by the size of some layer, both of which are bounded by the diameter of the graph. But the diameter of a real social networks is unlikely to grow linearly in the number of nodes and may even be viewable as a constant, as in the 6 degrees of separation phenomenon (e.g., [40]). Moreover, if the number of strongly connected components is $O(\log n)$ as the number of vertices n tends to infinity, then Solution (2.7) calls for a quadratic number of packets (in the number of vertices) to be used in the worst case. We also cannot directly compare the number of packets used in this model to preceding influence maximization frameworks because the bias values reflect that one packet is not generally adequate to effect the desired change, though the simulation results suggest that in practice the number of packets needed in this model is a constant multiple of the number of nodes in the network. Nonetheless, additional simulations are needed to further investigate parameters' effect on network outcomes.

Moreover, additional heuristics can be considered for the stochastic case of Info Bal, especially optimizing the order in which packets are seeded to nodes within a strongly connected component; perhaps nodes with large out-degree or expected times a seed packet is transmitted therefrom should be considered first as to minimize the number of packets sent to previous nodes (beyond what was prescribed when we attempted to neutralize its bias based on the preceding nodes). An integer quadratic programming approach may also be taken to minimizing l_2^2 given its above decomposition, and an integer linear

programming approach for l_1, using standard tricks for converting the absolute value into linear constraints. However, given that these problems are NP-hard even for simple special cases, they seem unlikely to be feasible for non-small-scale social networks. The simulation results, while robust in terms of an empirical approximation ratio, sometimes still result in large bias values even per node, which may not have much practical benefit. Another goal, albeit partially captured by l_μ, would be to ensure the bias values of sufficiently many or a certain subset of the nodes is small, though most formulations of this problem would likely be NP-hard (in fact, exact cover by 3-sets would still be reducible to such a problem by only considering subsets completely contained in the desired subset, the rest isolated with bias 0).

The results for the Const Info Bal simulations suggest that the effect of rounding the number of packets as in Equation (2.13) may become negligible in the presence of many independent bias values (as in the above simulation), in that the deviation in $+1$ and -1 packets reaching most nodes is close to 0. Therefore, that algorithm may admit an average-case approximation ratio.

Chapter 3: Ranking with Partitioning

Our treatment of ranking leaves several opportunities to strengthen the results presented therein.

For instance, the approximation guarantees may be able to be improved with the additional settings of the above special cases. The $\frac{k+2}{3}$-approximation algorithm for k-Set Packing holds for an arbitrary set-system with n subsets; it employs a local search algorithm with swaps of size $O(\log n)$. Assuming the structure of a graph—or in particular a $(k-1)$-regular graph where a vertex and its neighbors always form a k-component—may affect the guarantees possible with local search. It is worth noting that the authors responsible for the above guarantee do show that it is essentially sharp for local search algorithms that consider swap sizes that are linear in n, but the example is not an instance of the uniform value case of Rank Maximization [36]. Preliminary results suggest that, in a $(k-1)$-regular graph, local search may be used to obtain a constant fraction of the total possible $\lfloor \frac{n}{k} \rfloor$ disjoint k-components that may be in a graph. It also appears that applying the probabilistic method to randomly place vertices in subgraphs of size ck for some $c \in \{2, 3, \dots\}$ may be used to show the existence of a constant fraction of $\lfloor \frac{n}{k} \rfloor$ disjoint k-components in any $(k-1)$-regular graph. Nonetheless, some additional care is required to handle the conditional probabilities involved.

Similarly, exploring a special case of the partition problems where the input graph is planar would enable us to investigate perennial ordinal statements involving geography, e.g., that California has the 5th largest economy in the world [28]. This claim is based on treating California as a country (along with the remainder of the United States) yet otherwise maintaining the list of recognized countries by the International Monetary Fund, which already involves counterfactual reasoning that may be taken to its limit via the above framework.

Further bounds on the maximum and minimum rank may be pursued in more of the special cases and variants presented above. This is consistent with the overarching goal of the project: to study how robust rank and percentile are to partitioning by (potentially bad-faith) actors. Of particular interest are frequently encountered claims about societal problems made by news media or corporations. This includes analysis of social data in the United States by statistical area, Netflix's data of most-watched series by season, and nutritional labels for processed foods.

Chapter 4: Redistricting with Hierarchical Rectangles

The condition of population regularity (PR) is intended to allow a degree of flexibility consistent with the very granular notion of precincts in, say, United States redistricting contexts, even in the presence of a discrete model. This applies to our model of redistricting as a game as well: no player will ever get stuck after some number of cuts. We saw that (PR) was not enough to prevent a restriction to valid rectangle redistrictings from resulting in the worst-case deviation from proportional representation. Nonetheless, introducing (PR) to the construction separating valid rectangle redistrictings and hierarchical rectangle redistrictings presents several complications, allowing Player 1 to mix and match portions of different gadget grids in order to isolate a winning district, as confirmed by execution of the algorithm for Gerrymandering with Hierarchical Rectangles. Similarly, the worst-case outcome in the redistricting game was based on Player 1 being able to crack Player 2's support in the same pattern throughout the grid, whose coarse size (relative to the number of districts) allowed the first cut to impose the same cut orientation throughout the rest of the grid. Adding (PR) would let Player 2 choose whichever cut orientation they would prefer in each turn, complicating the design of any precinct loyalty function that would necessarily favor Player 1. Not only does including (PR) break both of the latter constructions, we conjecture that it precludes the worst-case gaps with proportional repre-

sentation in those two models, mitigating the unintended consequences of the restrictiveness that hierarchical rectangles impose.

Conversely, it is important to analyze the upsides of this restrictiveness in constraining gerrymandering, which we initiated with our urban-rural model. While allowing $C > \frac{L+W}{2}$, or equivalently $p_1 > .5$, adds additional cases to our calculations of p_1 and F, the lower bound on A still stands. Moreover, A and p_1 are continuous in C almost everywhere (and generally at the point $C = \frac{L+W}{2}$), so there are likewise examples where Player 1 has a slight majority and is still only able to gerrymander their proportional share of the eventual redistricting. On a theoretical, and especially a contemporary, level, there is interest in analyzing Player 2's gerrymandering power as well, who represents a rural party that might control the map making process. This, however, is not as straightforward as for Player 1. In particular, the condition $x_c + y_c \leq C$ on the centroid (x_c, y_c) of a rectangle district is a sufficient but not necessary for loyalty to Player 1. The reason is that the precinct loyalty function $p_1(\cdot, \cdot)$ "levels off" at a minimum value in the rural region of the grid, whereas Player 1's support keeps on increasing as the urban core is approached. Because our population-based cartogram approach inflates the size of the urban region, it allows Player 1 to control more of the geography when $p_1 > .5$. Contiguity may be thus be violated in some instances, which is still an issue for other grid-based models with uniform population. Additionally, though the ensuing districts would not generally be rectangles in a traditional map, the framing of hierarchical rectangles applied to a cartogram is still a potential technique to combat traditional power imbalances in gerrymandering ability for urban and rural parties.

APPENDIX A

List of CRT Categories

The following list gives the hierarchical breakdown of CRT Categories, with 2022-level emissions, in million metric tons of CO_2 equivalent, shown in parentheses. Indentation indicates a subcategory of the category immediately to the left.

All (6343.21)
- 1 Energy (5187.08)
  - 1.A (4854.82)
    - 1.A.1 (1554.53)
      - 1.A.1 Stationary Combustion - Coal - Electricity Generation (869.853)
        - 1.A.1 Stationary Combustion - Coal - Electricity Generation CH_4 (0.228783)
        - 1.A.1 Stationary Combustion - Coal - Electricity Generation CO_2 (851.472)
        - 1.A.1 Stationary Combustion - Coal - Electricity Generation N_2O (18.1526)
      - 1.A.1 Stationary Combustion - Geothermal Energy CO_2 (0.380557)
      - 1.A.1 Stationary Combustion - Natural Gas - Electricity Generation (663.75)
        - 1.A.1 Stationary Combustion - Natural Gas - Electricity Generation CH_4 (1.03868)
        - 1.A.1 Stationary Combustion - Natural Gas - Electricity Generation CO_2 (659.305)
        - 1.A.1 Stationary Combustion - Natural Gas - Electricity Generation N_2O (3.40572)
      - 1.A.1 Stationary Combustion - Oil - Electricity Generation (20.5295)
        - 1.A.1 Stationary Combustion - Oil - Electricity Generation CH_4 (0.00149642)
        - 1.A.1 Stationary Combustion - Oil - Electricity Generation CO_2 (20.5222)

- 1.A.1 Stationary Combustion - Oil - Electricity Generation N_2O (0.00580066)
- 1.A.1 Stationary Combustion - Wood - Electricity Generation (0.0191025)
  - 1.A.1 Stationary Combustion - Wood - Electricity Generation CH_4 (0.0018255)
  - 1.A.1 Stationary Combustion - Wood - Electricity Generation N_2O (0.017277)

1.A.2 (804.678)

- 1.A.2 Stationary Combustion - Coal - Industrial CO_2 (43.0351)
- 1.A.2 Stationary Combustion - Industrial (3.61476)
  - 1.A.2 Stationary Combustion - Industrial CH_4 (1.58172)
  - 1.A.2 Stationary Combustion - Industrial N_2O (2.03305)
- 1.A.2 Stationary Combustion - Natural Gas - Industrial CO_2 (510.385)
- 1.A.2 Stationary Combustion - Oil - Industrial CO_2 (247.644)

1.A.3 (1765.82)

- 1.A.3.a Transportation: Aviation (167.002)
  - 1.A.3.a Transportation: Aviation CH_4 (0.0367006)
  - 1.A.3.a Transportation: Aviation CO_2 (165.612)
  - 1.A.3.a Transportation: Aviation N_2O (1.35306)
- 1.A.3.b Transportation: Road (1447.94)
  - 1.A.3.b Transportation: Road CH_4 (0.910629)
  - 1.A.3.b Transportation: Road CO_2 (1438.14)
  - 1.A.3.b Transportation: Road N_2O (8.89341)
- 1.A.3.c Transportation: Railways (32.826)
  - 1.A.3.c Transportation: Railways CH_4 (0.0752349)
  - 1.A.3.c Transportation: Railways CO_2 (32.5229)
  - 1.A.3.c Transportation: Railways N_2O (0.227854)
- 1.A.3.d Transportation: Domestic Navigation (41.6155)
  - 1.A.3.d Transportation: Domestic Navigation CH_4 (0.469529)
  - 1.A.3.d Transportation: Domestic Navigation CO_2 (40.8857)
  - 1.A.3.d Transportation: Domestic Navigation N_2O (0.260323)
- 1.A.3.e Transportation: Other (76.4292)
  - 1.A.3.e Transportation: Other CH_4 (1.11726)
  - 1.A.3.e Transportation: Other CO_2 (69.3499)
  - 1.A.3.e Transportation: Other N_2O (5.96204)

1.A.4 (599.547)

- 1.A.4.a (260.473)
  - 1.A.4.a Stationary Combustion - Coal - Commercial CO_2 (1.38813)
  - 1.A.4.a Stationary Combustion - Commercial (1.73966)
    - 1.A.4.a Stationary Combustion - Commercial CH_4 (1.41059)
    - 1.A.4.a Stationary Combustion - Commercial N_2O (0.329069)
  - 1.A.4.a Stationary Combustion - Natural Gas - Commercial CO_2 (192.262)
  - 1.A.4.a Stationary Combustion - Oil - Commercial CO_2 (65.0824)
- 1.A.4.b (339.074)
  - 1.A.4.b Stationary Combustion - Coal - Residential CO_2 (0)
  - 1.A.4.b Stationary Combustion - Natural Gas - Residential CO_2 (271.987)
  - 1.A.4.b Stationary Combustion - Oil - Residential CO_2 (62.0785)
  - 1.A.4.b Stationary Combustion - Residential (5.00875)
    - 1.A.4.b Stationary Combustion - Residential CH_4 (4.30891)
    - 1.A.4.b Stationary Combustion - Residential N_2O (0.699838)
- 1.A.5 (130.245)
  - 1.A.5 Non-Energy Use of Fuels CO_2 (102.808)
  - 1.A.5 Stationary Combustion - Coal - U.S. Territories CO_2 (2.88857)
  - 1.A.5 Stationary Combustion - Natural Gas - U.S. Territories CO_2 (2.72723)
  - 1.A.5 Stationary Combustion - Oil - U.S. Territories CO_2 (16.9588)
  - 1.A.5 Stationary Combustion - U.S. Territories (0.0861103)
    - 1.A.5 Stationary Combustion - U.S. Territories CH_4 (0.0350626)
    - 1.A.5 Stationary Combustion - U.S. Territories N_2O (0.0510477)
  - 1.A.5.b Transportation: Military (4.77658)
    - 1.A.5.b Transportation: Military CH_4 (0.000279364)
    - 1.A.5.b Transportation: Military CO_2 (4.77618)
    - 1.A.5.b Transportation: Military N_2O (0.00012299)
- 1.B (332.264)
  - 1.B.1 (52.3999)
    - 1.B.1 Coal Mining CO_2 (2.47399)
    - 1.B.1 Fugitive Emissions from Abandoned Underground Coal Mines CH_4 (6.29926)
    - 1.B.1 Fugitive Emissions from Coal Mining CH_4 (43.6266)
  - 1.B.2 (279.864)
    - 1.B.2 Abandoned Oil and Natural Gas Wells (8.50287)
      - 1.B.2 Abandoned Oil and Natural Gas Wells CH_4 (8.49511)
      - 1.B.2 Abandoned Oil and Natural Gas Wells CO_2 (0.00775717)

- 1.B.2 Natural Gas Systems (209.733)
  - 1.B.2 Natural Gas Systems CH_4 (173.111)
  - 1.B.2 Natural Gas Systems CO_2 (36.47)
  - 1.B.2 Natural Gas Systems N_2O (0.151986)
- 1.B.2 Petroleum Systems (61.6286)
  - 1.B.2 Petroleum Systems CH_4 (39.6145)
  - 1.B.2 Petroleum Systems CO_2 (21.9666)
  - 1.B.2 Petroleum Systems N_2O (0.0475097)

2 Industrial Processes and Product Use (383.183)

- 2.A (66.4314)
  - 2.A.1 Cement Production CO_2 (41.8844)
  - 2.A.2 Lime Production CO_2 (12.2075)
  - 2.A.3 Glass Production CO_2 (1.9558)
  - 2.A.4 Other Process Uses of Carbonates CO_2 (10.3836)
- 2.B (77.5157)
  - 2.B.1 Ammonia Production CO_2 (12.6098)
  - 2.B.2 Nitric Acid Production N_2O (8.6125)
  - 2.B.3 Adipic Acid Production N_2O (2.0888)
  - 2.B.4 Caprolactam, Glyoxal, and Glyoxylic Acid Production N_2O (1.3356)
  - 2.B.5 Silicon Carbide Production and Consumption (0.222757)
    - 2.B.5 Silicon Carbide Production and Consumption CH_4 (0.012992)
    - 2.B.5 Silicon Carbide Production and Consumption CO_2 (0.209765)
  - 2.B.6 Titanium Dioxide Production CO_2 (1.474)
  - 2.B.7 Soda Ash Production CO_2 (1.70399)
  - 2.B.8 Petrochemical Production (28.7928)
    - 2.B.8 Petrochemical Production CH_4 (0.004788)
    - 2.B.8 Petrochemical Production CO_2 (28.788)
  - 2.B.9 Fluorochemical Production PFC, HFC, SF_6, NF_3 (7.7828)
  - 2.B.10 (12.8927)
    - 2.B.10 Carbon Dioxide Consumption CO_2 (5)
    - 2.B.10 Phosphoric Acid Production CO_2 (0.84009)
    - 2.B.10 Urea Consumption for Non-Ag Purposes CO_2 (7.05256)
- 2.C (46.7469)
  - 2.C.1 Iron and Steel Production & Metallurgical Coke Production (40.6796)
    - 2.C.1 Iron and Steel Production & Metallurgical Coke Production CH_4 (0.00771267)

- 2.C.1 Iron and Steel Production & Metallurgical Coke Production CO_2 (40.6719)
- 2.C.2 Ferroalloy Production (1.33736)
  - 2.C.2 Ferroalloy Production CH_4 (0.0104055)
  - 2.C.2 Ferroalloy Production CO_2 (1.32695)
- 2.C.3 Aluminum Production (2.20261)
  - 2.C.3 Aluminum Production CO_2 (1.44634)
  - 2.C.3 Aluminum Production PFCs (0.756268)
- 2.C.4 Magnesium Production and Processing (1.15229)
  - 2.C.4 Magnesium Production and Processing CO_2 (0.00294143)
  - 2.C.4 Magnesium Production and Processing HFCs (0.0288021)
  - 2.C.4 Magnesium Production and Processing SF_6 (1.12054)
- 2.C.5 Lead Production CO_2 (0.4275)
- 2.C.6 Zinc Production CO_2 (0.947466)

2.E Electronics Industry (4.73432)
- 2.E Electronics Industry N_2O (0.295248)
- 2.E Electronics Industry PFC, HFC, SF_6, NF_3 (4.43907)

2.F (178.134)
- 2.F.1 Emissions from Substitutes for Ozone Depleting Substances: Refrigeration and Air conditioning HFCs, PFCs (144.637)
- 2.F.2 Emissions from Substitutes for Ozone Depleting Substances: Foam Blowing Agents HFCs, PFCs (11.6871)
- 2.F.3 Emissions from Substitutes for Ozone Depleting Substances: Fire Protection HFCs, PFCs (2.64203)
- 2.F.4 Emissions from Substitutes for Ozone Depleting Substances: Aerosols HFCs, PFCs (17.0371)
- 2.F.5 Emissions from Substitutes for Ozone Depleting Substances: Solvents HFCs, PFCs (2.13094)

2.G (9.62075)
- 2.G Electrical Equipment PFC, SF_6 (5.07839)
- 2.G Other Product Manufacture and Use (4.54236)
  - 2.G Other Product Manufacture and Use N_2O (3.7503)
  - 2.G Other Product Manufacture and Use PFC, HFC, SF_6 (0.792061)

3 Agriculture (593.383)

3.A (192.578)
- 3.A.1 Enteric Fermentation: Cattle CH_4 (185.9)

- 3.A.4 Enteric Fermentation: Other Livestock CH_4 (6.67784)
- 3.B (81.7182)
  - 3.B.1 Manure Management: Cattle (50.3113)
    - 3.B.1 Manure Management: Cattle CH_4 (37.7265)
    - 3.B.1 Manure Management: Cattle N_2O (12.5848)
  - 3.B.4 Manure Management: Other Livestock (31.4069)
    - 3.B.4 Manure Management: Other Livestock CH_4 (26.998)
    - 3.B.4 Manure Management: Other Livestock N_2O (4.40883)
- 3.C Rice Cultivation CH_4 (18.8673)
- 3.D (290.802)
  - 3.D.1 Direct Agricultural Soil Management N_2O (262.477)
  - 3.D.2 Indirect Applied Nitrogen N_2O (28.3245)
- 3.F Field Burning of Agricultural Residues (0.822987)
  - 3.F Field Burning of Agricultural Residues CH_4 (0.620643)
  - 3.F Field Burning of Agricultural Residues N_2O (0.202344)
- 3.G Liming CO_2 (3.268)
- 3.H Urea Fertilization CO_2 (5.32742)

5 Waste (179.56)

- 5.A (119.767)
  - 5.A Commercial Landfills CH_4 (100.856)
  - 5.A Industrial Landfills CH_4 (18.9112)
- 5.B (4.42602)
  - 5.B Composting (4.41264)
    - 5.B Composting CH_4 (2.58076)
    - 5.B Composting N_2O (1.83188)
  - 5.B.2 Anaerobic Digestion at Biogas Facilities CH_4 (0.0133801)
- 5.C.1 Incineration of Waste (12.691)
  - 5.C.1 Incineration of Waste CH_4 (0.000140888)
  - 5.C.1 Incineration of Waste CO_2 (12.3575)
  - 5.C.1 Incineration of Waste N_2O (0.333344)
- 5.D (42.6765)
  - 5.D Domestic Wastewater Treatment (34.9967)
    - 5.D Domestic Wastewater Treatment CH_4 (13.5907)
    - 5.D Domestic Wastewater Treatment N_2O (21.406)
  - 5.D Industrial Wastewater Treatment (7.67978)
    - 5.D Industrial Wastewater Treatment CH_4 (7.21399)

5.D Industrial Wastewater Treatment N_2O (0.465789)

BIBLIOGRAPHY

[1] Rediet Abebe. *Designing Algorithms for Social Good*. PhD thesis, Cornell University, 2019.

[2] U.S. Environmental Protection Agency. Inventory of u.s. greenhouse gas emissions and sinks: 1990-2022. Technical report, U.S. Environmental Protection Agency, 2024.

[3] Noga Alon and Joel H Spencer. *The probabilistic method*. John Wiley & Sons, 2016.

[4] Georgios Amanatidis, Haris Aziz, Georgios Birmpas, Aris Filos-Ratsikas, Bo Li, Hervé Moulin, Alexandros A. Voudouris, and Xiaowei Wu. Fair division of indivisible goods: Recent progress and open questions. *Artificial Intelligence*, 322:103965, 2023.

[5] 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 Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 98–104. International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[6] Isabelle Brocas and Juan D. Carrillo. Influence through ignorance. *The RAND Journal of Economics*, 38(4), 2007.

[7] Eric Bryant. New england minor civil divisions. License: CC BY-SA 4.0.

[8] Adam L. Buchsbaum, Emden R. Gansner, Cecilia M. Procopiuc, and Suresh Venkatasubramanian. Rectangular layouts and contact graphs. *ACM Trans. Algorithms*, 4(1), March 2008.

[9] John W. Byers, Michael Mitzenmacher, and Georgios Zervas. The groupon effect on yelp ratings: A root cause analysis. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, EC '12, page 248–265, New York, NY, USA, 2012. Association for Computing Machinery.

[10] Tim Carnes, Chandrashekhar Nagarajan, Stefan M. Wild, and Anke van Zuylen. Maximizing influence in a competitive social network: A follower's perspective. In *Proceedings of the Ninth International Conference on Electronic Commerce*, ICEC '07, page 351–360, New York, NY, USA, 2007. Association for Computing Machinery.

[11] Ward Casscells, Arno Schoenberger, and Thomas B. Graboys. Interpretation by physicians of clinical laboratory results. *New England Journal of Medicine*, 299(18):999–1001, 1978. PMID: 692627.

[12] Nicolas Champseix, Esther Galby, Andrea Munaro, and Bernard Ries. Cpg graphs: Some structural and hardness results. *Discrete Applied Mathematics*, 290:17–35, 2021.

[13] C. C. Chen and N. F. Quimpo. On strongly hamiltonian abelian group graphs. In Kevin L. McAvaney, editor, *Combinatorial Mathematics VIII*, pages 23–34, Berlin, Heidelberg, 1981. Springer Berlin Heidelberg.

[14] Sijing Chen, Lu Xiao, and Akit Kumar. Spread of misinformation on social media: What contributes to it and how to combat it. *Computers in Human Behavior*, 141:107643, 2023.

[15] Wei Chen, Alex Collins, Rachel Cummings, Te Ke, Zhenming Liu, David Rincon, Xiaorui Sun, Yajun Wang, Wei Wei, and Yifei Yuan. *Influence Maximization in Social Networks When Negative Opinions May Emerge and Propagate*, pages 379–390.

[16] Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, page 57–66, New York, NY, USA, 2001. Association for Computing Machinery.

[17] M.E. Dyer and A.M. Frieze. On the complexity of partitioning graphs into connected subgraphs. *Discrete Applied Mathematics*, 10(2):139–153, 1985.

[18] Gabriele Esposito, Penélope Hernández, René van Bavel, and José Vila. Nudging to prevent the purchase of incompatible digital products online: An experimental study. *PLoS One*, 12(3):e0173333, 2017.

[19] Jason Gaitonde, Jon Kleinberg, and Éva Tardos. Polarization in geometric opinion dynamics. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, EC '21, page 499–519, New York, NY, USA, 2021. Association for Computing Machinery.

[20] Waldo Gálvez, Arindam Khan, Mathieu Mari, Tobias Mömke, Madhusudhan Reddy, and Andreas Wiese. A (2+ϵ)-approximation algorithm for maximum independent set of rectangles, 2021.

[21] Jan Hązła, Yan Jin, Elchanan Mossel, and Govind Ramnarayan. A Geometric Model of Opinion Polarization. Papers 1910.05274, arXiv.org, October 2019.

[22] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, page 137–146, New York, NY, USA, 2003. Association for Computing Machinery.

[23] David G. Kirkpatrick and Pavol Hell. On the completeness of a generalized matching problem. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, STOC '78, page 240–245, New York, NY, USA, 1978. Association for Computing Machinery.

[24] Sébastien Lahaie, David M. Pennock, Amin Saberi, and Rakesh V. Vohra. *Sponsored Search Auctions*, page 699–716. Cambridge University Press, 2007.

[25] Jure Leskovec and Andrej Krevl. SNAP Datasets: Gnutella peer-to-peer network, august 6 2002. Stanford Large Network Dataset Collection, https://snap.stanford.edu/data/p2p-Gnutella06.html, August 2014.

[26] Michigan Bureau of Elections. *Election Officials Manual Chapter 15: Establishing Voting Precincts, Polling Places, and Early Voting Sites.* Michigan Secretary of State, Lansing, MI, July 2024.

[27] Stephanie Nolen. As vaccines trickle into africa, zambia's challenges highlight other obstacles. *The New York Times*, 2021.

[28] Office of Governor Gavin Newsom. California remains the world's 5th largest economy. https://www.gov.ca.gov/2024/04/16/california-remains-the-worlds-5th-largest-economy/, April 2024.

[29] Tore Opsahl and Pietro Panzarasa. Clustering in weighted networks. *Social Networks*, 31(2):155–163, 2009.

[30] Wesley Pegden, Ariel D Procaccia, and Dingli Yu. A partisan districting protocol with provably nonpartisan outcomes. *arXiv preprint arXiv:1710.08781*, 2018.

[31] Ivens Portugal, Paulo Alencar, and Donald Cowan. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*, 97:205–227, 2018.

[32] J. Scott Provan and Michael O. Ball. The complexity of counting cuts and of computing the probability that a graph is connected. *SIAM Journal on Computing*, 12(4):777–788, 1983.

[33] Matthew Richardson and Pedro Domingos. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, page 61–70, New York, NY, USA, 2002. Association for Computing Machinery.

[34] C.S. Rim and K. Nakajima. On rectangle intersection and overlap graphs. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 42(9):549–553, 1995.

[35] Susmita Sur-Kolay and Bhargab B. Bhattacharya. Inherent nonslicibility of rectangular duals in vlsi floorplanning. In *Proceedings of the Eighth Conference on Foundations of Software Technology and Theoretical Computer Science*, page 88–107, Berlin, Heidelberg, 1988. Springer-Verlag.

[36] Maxim Sviridenko and Justin Ward. Large neighborhood local search for the maximum set packing problem. In *Automata, Languages, and Programming: 40th International Colloquium, ICALP 2013*, volume 7965 of *Lecture Notes in Computer Science*, pages 792–803. Springer, 2013.

[37] United States Census Bureau. New england city and town areas wall map (March 2020), March 2020.

[38] U.S. Environmental Protection Agency. Sources of greenhouse gas emissions, 2025. Accessed: April 17, 2026.

[39] U.S. Food and Drug Administration. 21 cfr 101.4. Electronic Code of Federal Regulations (eCFR), December 2024.

[40] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.