Title
Algorithmic Problems in Committee Selection

Author
Sonar, Chinmay

Publication Date
2023

University of California
Santa Barbara

Algorithmic Problems in Committee Selection

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Computer Science

by

Chinmay Sonar

Committee in charge:

Professor Subhash Suri, Chair
Professor Daniel Lokshtanov
Professor Eric Vigoda

December 2023

The Dissertation of Chinmay Sonar is approved.

Professor Daniel Lokshtanov

Professor Eric Vigoda

Professor Subhash Suri, Committee Chair

December 2023

Algorithmic Problems in Committee Selection

by

Chinmay Sonar

To my parents

Acknowledgements

I begin by thanking my advisor Subhash Suri for introducing me to computational geometry research and geometric committee selection and helping me select an amazing committee for my thesis. I want thank him even more for giving me immense independence in choosing my research problems and allowing me to set my goals and my pace through the research process, and still be available for guidance whenever I need. I leaned a lot through his profound simplicity in finding clean research questions and clarity in his writing. I am indebted to Daniel Lokshtanov not only for the research guidance but also for encouraging us to do a lot of physical activities and keep in shape. Thanks to him also for teaching me many new concepts not only in CS theory research but also in skiing, trail running and the life in general. I am fortunate to meet a friend like Daniel, and the work in this thesis would not have been so fulfilling without his perpetual energy and optimism. I also want to thank Eric Vigoda for introducing me to the field of combinatorial counting and for helping me write a good thesis and develop my presentation skills. Thanks also for inviting many many (celebrity) guest speakers and arranging meetings with them, talking to them really helped me shape my perspective of CS research.

A special thanks for Jie Xue for all his support and guidance throughout my PhD. I was very fortunate to work with him since the very outset of my graduate school journey, and I learned a lot of new concepts and writing skills from him. Thanks to him for his quarter long visit during my last quarter at UCSB, I really enjoyed our in-person discussions. A big thanks also to Neeldhara Misra for her unwavering support and encouragement throughout my PhD. Her belief in me and her guidance has been invaluable and it always keeps me motivated to become better academically and personally.

I am thankful to all the CS department faculties for teaching such amazing classes and providing amicable interdisciplinary research environment. Mentioning everyone is going to be hard but I would like to highlight my interactions with Diba Mirza, William Wang, Ambuj Singh and Divyakant Agrawal, and thank all of them for their support.

I am extremely grateful to all my friends for making this journey fulfilling. I am so very fond of all the friends I have gathered at UCSB and the friends who continued to be together from before, and I consider myself fortunate to be associated with such amazing human beings. A special thanks to Úrsula who was my labmate, housemate, climbing/hiking companion and a close friend for sharing many many fun moments together. Thanks to Kalyan, Subramaniyam, Swaroop, Radha, Vaishali, and Sesha for arranging so many trips, pursuing so many discussions, and most importantly being there for me through the ups and downs of life during my PhD. Thanks also to Amogh, Harsh and Pranali for arranging a memorable trip to the east coast. Thanks to Vismay, Vaidyanathan, Kshiteej, Sanket, Aditya, and Pinank for passing the test of time and still keeping in touch despite being separated by timezones.

Finally, I would like to thank my parents, my sister and my extended family who are truly the wind under my wings; their unconditional trust and support means the world. Thanks also to Vikrant, Saurabh, Kaustubh and Ketaki for making all of my India trips special. I am short of words to thank my family enough for their invaluable presence my life, my memories, and my whole essence, hence, I would like to be indebted to them for my lifetime.

Curriculum Vitæ

Chinmay Sonar

Education

2023	Ph.D. in Computer Science (Expected), University of California, Santa Barbara.
2019	MTech in Computer Science, Indian Institute of Technology Gandhinagar.
2019	BTech in Mechanical Engineering, Indian Institute of Technology Gandhinagar.

Publications

1. **Fault-Tolerance in Euclidean Committee Selection**
   with Subhash Suri and Jie Xue.
   In *European Symposium on Algorithms (ESA'23), Amsterdam, Netherlands.*
2. **Multiwinner Elections under Minimax Chamberlin-Courant Rule in Euclidean Space**
   with Subhash Suri and Jie Xue.
   In *International Joint Conference on Artificial Intelligence (IJCAI'22), Vienna, Austria.*
3. **Anonymity-Preserving Space Partitions**
   with Úrsula Hébert-Johnson, Subhash Suri and Vaishali Surianarayanan.
   In *International Symposium on Algorithms and Computation (ISAAC'21), Fukuoka, Japan.*
4. **Equitable Division of a Path**
   with Neeldhara Misra, P. R. Vaidyanathan and Rohit Vaish.
   In *COMSOC 2021*
5. **Fair Covering of Points by Balls**
   with Daniel Lokshtanov, Subhash Suri and Jie Xue.
   In *Canadian Conference of Computational Geometry (CCCG 2020), Saskatoon, Canada.*
6. **On the Complexity of Winner Verification and Candidate Winner for Multiwinner Voting Rules**
   with Palash Dey and Neeldhara Misra.
   In *International Joint Conference on Artificial Intelligence (IJCAI'20), Yokohama, Japan.*
7. **Robustness Radius for Chamberlin-Courant on Restricted Domains**
   with Neeldhara Misra.
   In *International Conference on Current Trends in Theory and Practice of Informatics (SOFSEM'19), Nový Smokovec, Slovakia.*

8. **On the Complexity of Chamberlin-Courant On Almost Structured Profiles**
   with Neeldhara Misra and P. R. Vaidyanathan.
   In *International Conference on Algorithmic Decision Theory (ADT'17), Luxembourg.*

9. **Let's Think Frame by Frame: Evaluating Video Chain of Thought with Video Infilling and Prediction**
   with Vaishnavi Himakunthala, Andy Ouyang, Daniel Rose, Ryan He, Alex Mei, Yujie Lu, Michael Saxon, William Yang Wang.
   In *The 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP'23)*, Singapore

10. **Visual Chain of Thoughts: Bridging Logical Gaps with Multimodal Infillings**
    with Daniel Rose, Vaishnavi Himakunthala, Andy Ouyang, Ryan He, Alex Mei, Yujie Lu, Michael Saxon, Diba Mirza, William Yang Wang.
    In submission to ICLR'24.

Work Experience

June - September 2023	Visa, Inc.	ML Scientist (PhD Intern), Palo Alto, CA
June - September 2022	Visa, Inc.	ML Scientist (PhD Intern), Austin, TX

Miscellaneous

- **Teaching Assistant**
  - **UCSB:** CS 130A, 130B (Data Structures and Algorithms), CS 24 (Problem Solving with Computers), CS 8 (Introduction to Python Programming)
  - **IIT Gandhinagar:** CS 112 (Introduction to Python Programming), ES 214 (Discrete Mathematics)
- **Undergraduate Mentoring (UCSB)**
  - Central mentor for the Early Research Scholar Program (ERSP) for two years (mentored 40 sophomores each year on how to get started with research in CS)
  - Research project mentor (mentored a group of 4 people in a yearlong research project 2022)
- **Program Committee and External Reviewer for Conferences**
  - PC member for AAMAS'24
  - External Reviewer for STACS'24
  - External Reviewer for SODA'20

- External Reviewer for MFCS'19
  - External Reviewer for AAMAS'19
- **External Reviewer for Journals**
  - Transactions on Intelligent Systems and Technology (TIST)
  - Transactions on Knowledge Discovery from Data (TKDD)

Abstract

Algorithmic Problems in Committee Selection

by

Chinmay Sonar

Committee selection is a classical problem in the social sciences where the goal is to choose a fixed number of candidates based on voters' preferences. The problem naturally models elections in representative democracies or hiring of staff, but also generalizes many other resource allocation problems such as the classical facility location problems where facilities are treated as candidates and users as voters. By explicitly modeling voters' preferences, the committee selection problem raises a number of interesting and challenging algorithmic problems such as winner determination, fault tolerance, and fairness. In this dissertation, we study four such problems.

For the most part, we consider the committee selection problem in Euclidean d-space where candidates/voters are points and voters' preferences are implicitly derived using their Euclidean distances to the candidates. Our first problem is to find a winning committee under the well-known Chamberlin-Courant voting rule. The goal here is to choose a committee of k candidates so that the rank of any voter's most preferred candidate in the committee is minimized. (The problem is also equivalent to the ordinal version of the classical k-center problem.) We show that the problem is NP-hard in any dimension $d \geq 2$, and is also hard to approximate. Our main results are three polynomial-time approximation schemes, each of which finds a committee with a good minimax score.

Our second problem deals with fault tolerance in committee selection. We study the following three variants: (1) given a committee and a set of f failing candidates, find their

optimal replacement; (2) compute the worst-case replacement score for a given committee under failure of f candidates; and (3) design a committee with the best replacement score under worst-case failures. Our main results are polynomial-time algorithms for three problems in one dimension. We also show that the problems are NP-hard in higher dimensions and give constant-factor approximations for all three problems along with an FPT bicriterion approximation for the optimal committee problem.

In our third problem we consider non-Euclidean elections and study the following two natural questions for a given election: (1) (Winner Verification) Given a subset of candidates (committee) T, is T a winning committee? (2) (Candidate Winner) Given a candidate c, does c belong to a winning committee? We show that both the above problems are hard (coNP-complete and θ_2^P-complete, respectively) in general, but for the restricted case of single-peaked and single-crossing preferences, they admit efficient algorithms.

Our last problem is the problem of covering a multicolored set of points in $\mathbb{R}^d$ using (at most) k disjoint unit-radius balls chosen from a candidate set of unit-radius balls so that each color class is covered fairly in proportion to its size. Specifically, we investigate the complexity of covering the maximum number of points in this setting. In the committee selection terminology, each ball is a candidate and each point is a voter; a voter approves all candidates within a unit-radius ball around it, and our goal is to choose an optimal size k committee under fairness constraints. We show that the problem is NP-hard even in one dimension when the number of colors is not fixed. On the other hand, for a fixed number of colors, we present a polynomial-time exact algorithm in one dimension, and a PTAS in any fixed dimension $d \geq 2$.

Contents

Curriculum Vitae **vii**

Abstract **x**

1 Introduction **1**
1.1 Problems Studied and Our Contributions 3
1.2 Organization of Chapters 12
1.3 Permissions and Attributions 13

2 Winner Determination Under the Chamberlin-Courant Rule **15**
2.1 Hardness Results 18
2.2 Approximation using epsilon-nets 32
2.3 Bicriterion Approximation by Relaxing the Committee Size 38
2.4 Approximation by combining Cardinal and Ordinal Score 41
2.5 Bibliographic notes 46

3 Fault-Tolerant Committee Selection (FTCS) **47**
3.1 Optimal Replacement Problem 50
3.2 Fault-Tolerance Score 51
3.3 Optimal Fault-Tolerance Committee 55

4 FTCS in Multidimensional Instances **59**
4.1 NP-hardness Results 60
4.2 Constant factor Approximations 74
4.3 Bicriterion Approximation Scheme 82
4.4 Bibliographic notes 90

5 Committee Winner Verification Problem **92**
5.1 Chamberlin-Courant Voting Rule 99
5.2 Monroe Voting Rule 103
5.3 Elections with Approval Preferences 108
5.4 Efficient Algorithms on Restricted Preferences 110

6 Candidate Winner Verification Problem **113**
6.1 Chamberlin-Courant Voting Rule 115
6.2 Monroe Voting Rule 122
6.3 Elections with Approval Preferences 127
6.4 Efficient Algorithms on Restricted Preference 129

7 Fair Covering of Points in Euclidean Space **135**
7.1 Polynomial time Algorithm in One-dimension 138
7.2 NP-hardness and W[1]-hardness of Fair Covering 143
7.3 PTAS using Shifting Technique 145
7.4 Unconstrained Fair Covering in 1D 153
7.5 Bibliographic Notes 156

8 Conclusion and Open Problems **157**

Bibliography **161**

Chapter 1

Introduction

Consider the following city planning scenario: The municipal council has allocated some funds to build recreational venues in a district in the city. There are several options such as a convention center, a conference hall, a museum or a zoo, a golf course, a stadium or a climbing gym, and many more, but the allocated funds will only be sufficient for three new facilities. In order to choose which facilities to build, the council wants to do *participatory planning* where the entire local community is involved in the district planning process. Individuals in the local community have different preferences, the goal of the council is to elicit these preferences and then use them to make the final choice of three facilities such that the choice satisfies as many individuals as possible. In abstract terms, in the above scenario we have a limited public resource that can be spent on a few possible options, and different stakeholders have preferences over these options; the goal is to use their preferences to find a small subset of options to spend the resource on such that the stakeholders are as "happy" as possible. We call this the *Committee Selection Problem.* This abstraction captures many other interesting scenarios such as choosing representatives in a democracy, staff hiring and procurement decisions, shortlisting candidates for a limited fellowship, jury selection, cache management, etc.

[1, 2, 3, 4, 5, 6, 7]. These scenarios are ubiquitous and choosing the right set of candidates is crucial; hence studying this abstract problem is important. This thesis focuses on a few computational aspects of the committee selection problem.

Formally, the committee selection problem can be described as follows: We have a set $\mathcal{C}$ of m candidates, a set V of n voters and a committee size k. The goal is to choose a subset of k candidates denoted by $T \subseteq \mathcal{C}$ called a *winning committee*, that collectively best represents the preferences of all the voters. To relate the above abstract setting to the formal definition, observe that the set of possible options is the set of candidates, the stakeholders are the voters and the limited resource (budget) is the fixed committee size k. Every committee T has an associated score computed by a predetermined function f (the function f takes voters' preferences and T as input, and returns an integer score) which quantifies the "goodness" of T. Given candidates, voters, and preferences, the main problem is to efficiently find a committee of size k that optimizes the score. In this thesis, we focus on the computational aspects of the committee selection problems. That is, our aim to design efficient algorithms [1] which can compute an optimal (i.e., winning) committee as quickly as possible.

The committee selection problem as described above is known to be NP-hard. It is believed that NP-hard problems cannot be solved optimally in polynomial time (for more details, refer to [9, 10]). A super-polynomial running time is quite slow for practical applications; therefore, researchers have studied *approximation algorithms* where we settle for a near optimal solution to get faster (possibly polynomial-time) algorithms. Another way to obtain faster algorithms is to consider special/structured cases (i.e., restricted input instances) of the general problem. Considering special cases is a good research direction since the real-life problem instances are usually structured. Furthermore, in

[1] At a high level, an algorithm is a sequence of steps to solve a computational problem. Please find more details in [8].

many cases, the study of structured input instances helps us understand the general problem better. For the most part, in this thesis we take two approaches: the first is to design approximation algorithms for committee selection problems, and the second is to consider these problems on restricted input classes.

1.1 Problems Studied and Our Contributions

We also refer to the committee selection problem as the *multiwinner election* problem. We first describe Euclidean elections (which is a restricted input class) and then we introduce a committee selection rule (i.e., a voting rule) known as the Chamberlin-Courant.

Euclidean elections. We represent an election as $E = (\mathcal{C}, V)$, where $\mathcal{C}$ is the set of candidates and V is the set of voters. In a Euclidean election, candidates and voters are embedded in d-dimensional Euclidean space, and the preferences of the voters over the candidates are based on Euclidean distances, namely, a voter prefers a candidate closer to it over a farther candidate.

We consider the ranking (ordinal) preferences where each voter $v \in V$ ranks all candidates in $\mathcal{C}$ from the most preferred (rank 1) to the least preferred (rank m). For Euclidean elections, voters' rankings are implicitly defined by Euclidean distances (closest to farthest). Let $\sigma_v(c)$ denote the rank of candidate c in v's preference list. Next, we introduce the Chamberlin-Courant voting rule [11] which we use for evaluating the score of a committee.

Chamberlin-Courant Voting Rule. Under the Chamberlin-Courant rule, voter v's score for a committee T is the rank of its most preferred candidate: $\sigma_v(T) = \min_{c \in T} \sigma_v(c)$. In a sense, v is "assigned to" or "represented by" its top choice in the committee. The overall score of the committee T is some function $g(\sigma_{v_1}(T), \ldots, \sigma_{v_n}(T))$ of all the voter's scores. Two classical choices for g are the *sum* and the *max*. The former is the *utilitarian* objective and seeks to minimize the *sum* of the scores over all the voters (this min-sum aggregation is also referred to as ℓ_1 aggregation based on the 1-norm). The latter is the *egalitarian* objective and minimizes the *maximum* (worst) of the scores over all the voters (the min-max aggregation is also referred to as ℓ_∞ aggregation based on the ∞-norm). In Chapters 2, 3 and 4, we work with the egalitarian objective where the score of a committee is $\sigma(T) = \max_{v \in V} \sigma_v(T)$, and in Chapters 5 and 6, we work with both. Finally, in Chapter 7, we work only with the utilitarian objective for the case of *approval elections* which we introduce in Section 1.1.3.

We now present the problems studied in this dissertation and give a high-level idea of our contributions to them.

1.1.1 Winner Determination under the Chamberlin-Courant Rule

Our first problem is to compute a winning committee under the Chamberlin-Courant voting rule for Euclidean elections. Given a Euclidean election $E = (\mathcal{C}, V)$ and a committee size k, our goal here is to find a committee $T \subseteq \mathcal{C}$ of size k that optimizes (minimizes) the score $\sigma(T)$. We consider the ranking preferences and so the goal is to minimize the worst case representation rank, i.e., $\max_{v \in V} \sigma_v(T)$. Observe that this problem is equivalent to the ordinal version of the classical k-center problem [12]. (In the classical k-center problem, the score of a voter for a committee T is determined by its distance to the closest candidate in T rather than the rank of the most preferred candidate.)

For general preferences (unstructured preferences), finding a winning committee under Chamberlin-Courant is NP-hard for both the min-sum [5] and the min-max [1] objectives; as a result, an important line of research has been to examine natural settings with structured preferences [1, 13, 14, 15]. Our setting of Euclidean preferences is arguably the most natural setting. The geometry of Euclidean space gives an intuitive and interpretable *positioning* of voters and candidates in many natural settings such as spatial voting and facility locations, but it also has important computational advantages: when candidates and voters are embedded in d-space, only a tiny fraction of all (exponentially many) $m!$ candidate orderings are *realizable*. In particular, the maximum number of realizable rankings is only $\mathcal{O}(m^{d+1})$ (i.e., polynomially bounded). This important combinatorial property enables us to derive much better bounds than what is possible in completely unstructured preference spaces.

We first settle the complexity of the winner determination problem by showing that (unfortunately) the problem still remains NP-hard even for Euclidean elections. We reduce from Planar Monotone 3-SAT which was introduced and shown to be NP-hard in [16]. By a slight modification of our NP-hardness proof, we can also show that even computing an approximate Euclidean minimax committee within any sublinear factor of $|\mathcal{C}|$ is hard. This result is in sharp contrast to the classical k-center problem for which a 2-approximation algorithm is known. Our hardness result for this problem is summarized in Theorem 2 as follows:

"**Theorem:** For any constant $\varepsilon > 0$, it is NP-hard to achieve a $|\mathcal{C}|^{1-\varepsilon}$-approximation for Euclidean minimax committee in $\mathbb{R}^d$ for any $d \geq 2$."

Next, we turn to approximation algorithms. Our first result is a polynomial time $\mathcal{O}(m/k)$ approximation in 2-dimensions and $\mathcal{O}((m/k)\log k)$approximation for $d \geq 3$. To obtain the above results, we use ε-nets to design an algorithm to compute a committee of size k with a minimax score $\mathcal{O}(m/k)$ for $d = 2$, and a committee of size $\mathcal{O}((m/k)\log k)$ for $d \geq 3$. For the special case when the candidate set is a subset of the voters, i.e., $\mathcal{C} \subseteq V$, we show that the optimal minimax score has a lower bound of $\Omega(m/k)$ in all instances using an argument based on kissing number [2]. By combining this lower bound

[2] Kissing number is defined as the greatest number of non-overlapping (closed) unit spheres that can

with the above approximation algorithm, we obtain a constant-factor approximation in 2 dimensions and a $\mathcal{O}(\log k)$ approximation for $d \geq 3$ when $\mathcal{C} \subseteq V$.

Our next approximation uses the bicriterion framework to design a polynomial-time algorithm that achieves the *optimal* minimax score $\sigma^\star$ for a committee of size k by constructing a slightly larger committee, namely, of size $(1+\varepsilon)k$ for $d = 2$ and size $\mathcal{O}(k \log m)$ for $d \geq 3$. We also show that increasing the committee size by an *additive* constant is not sufficient to bypass the inapproximability imposed by our hardness result.

In our final result, we combine the ordinal (rank-based) and cardinal (distance-based) features of the problem in a novel way to get an approximation algorithm. Suppose the optimal score of the size-k committee is $\sigma^\star$, and $d_v^\star$ is the distance from v to its rank-$\sigma^\star$ candidate. We define a committee T to be δ*-optimal* if each voter has a representative in T within distance $\delta d_v^\star$. We show a 3-approximation, that is, we give a polynomial-time algorithm that computes a 3-optimal committee, and we show that we cannot compute a δ-optimal committee for any $\delta < 2$ in polynomial time unless P = NP.

1.1.2 Fault-Tolerant Committee Selection

In our next problem, we consider fault tolerance in committee selection, that is, we want to quantify how robust a chosen committee is against the possibility that some of the winning committee members may become unavailable (*fail*). The question of fault tolerance in committee selection is well-motivated this is because faults are uncommon in many scenarios modeled by committee selection problems such as democratic elections, staff hiring, jury selection, etc. In our work, we consider three different fault-tolerant committee selection problems.

Consider a Euclidean election $E = (\mathcal{C}, V)$. For this problem, we consider the minimax Chamberlin-Courant rule with cardinal (distance based) score, that is, voter v's score for a

be arranged in the Euclidean space such that they each touch a common unit sphere.

committee $T \subseteq \mathcal{C}$ is: $\sigma_v(T) = \min_{c \in T} dist(v, c)$. Recall that the score of the committee T is: $\sigma(T) = \max_{v \in V} \sigma_v(T)$ the worst-case voter score. We denote a set of failing candidates by $J \subseteq \mathcal{C}$, and denote the integer fault-tolerance parameter by f. We will now describe our problems at a high-level (see Section 3.0.1 for the formal definitions):

- **Optimal Replacement Problem (ORP)**: Given an election $E = (\mathcal{C}, V)$, a committee T and a failing set J, the goal is to find an optimal replacement (of failed committee members) $R \subseteq \mathcal{C} \setminus J$ with $|R| \leq |T \cap J|$ which minimizes $\sigma(T \setminus J \cup R)$. That is, we study how badly the given failing set affects the score of the committee.

- **Fault-tolerance Score (FTS)**: Given $E = (\mathcal{C}, V)$, a committee T and the fault-tolerance parameter f, we compute the *effect of the worst-case* size at most f failing set of candidates on the score of T, i.e., for each failing set of size at most f, we compute the score of the optimal replacement and return the worst score overall.

- **Optimal Fault-Tolerance Committee (OFTC)**: Given $E = (\mathcal{C}, V)$, a committee size k and fault-tolerance parameter f, we want to *proactively* compute a size k committee with minimum worst case f-fault tolerance score.

We now summarize our results. In one-dimensional instances when candidates and voters are embedded on a line, we show the following:

"**Theorem:** ORP, FTS, and OFTCS can be solved in a polynomial time on one-dimensional instances."

We obtain the above result by combining Theorems 11, 12 and 13. For dimensions $d \geq 2$, we show the following in Theorem 14:

"**Theorem:** ORP, FTS, and OFTC are NP-hard in any dimension $d \geq 2$, when the committee size k and the fault-tolerance parameter f are part of the input."

Next, we turn to our algorithmic results. We give polynomial time constant factor approximation for all three problems with factors $3, 3$ and 5 for ORP, FTS and OFTC, respectively. For a special case when f is a constant, we give a 3-approximation in polynomial time for OFTC. All of the above constant factor approximations are based on clever packing arguments and hold in any metric space. Our main algorithmic result in dimensions $d \geq 2$ is a novel bicriterion EPTAS (Efficient Polynomial-Time Approximation Scheme) for OFTC where suppose the optimal sore for a size k committee is σ^*, then our algorithm returns a size k committee that achieves the score $(1+\varepsilon)\sigma^*$ for at least $(1-\varepsilon)$ fraction of the voters. We obtain the above EPTAS with a nontrivial application of the (well-known) grid shifting technique [17].

1.1.3 Winner Verification Problems

The next set of problems we consider are the winner verification problems. Given an election $E = (\mathcal{C}, V)$ and the committee size k, we consider the following two natural variants of this problem:

1. **Committee Winner Verification:** For a size k subset $T \subseteq \mathcal{C}$, is T a winning committee?

2. **Candidate Winner Verification:** For a candidate $c \in \mathcal{C}$, does there exists a size k winning committee containing c?

We consider these two questions on general (non-Euclidean) elections for both ranking preferences and approval ballots. Recall that in the case of rankings, each voter gives a complete preference ordering over the set of candidates. Suppose the number of candidates is m then in *approval ballots*, each voter gives an m-length binary vector denoting its approved/non-approved candidates. Consider an election with 5 candidates, then an example voter preference under rankings is $v := c_2 \succ c_4 \succ c_1 \succ c_5 \succ c_3$ and under approval ballots is $v = [1, 1, 0, 1, 0]$ which represents that v only approves candidates c_1, c_2 and c_4.

In this work, along with the Chamberlin-Courant rule, we also consider the *Monroe rule* [18] where the only difference from Chamberlin-Courant is that each committee member is assigned to the same number of voters. Observe that under the Monroe rule, proportional representation is explicitly ensured.

We consider both min-sum (ℓ_1) and min-max (ℓ_∞) aggregations and by reductions from the complement of hitting set problem we obtain the following result (Theorem 21) for problem 1:

"**Theorem:** Committee Winner Verification for Chamberlin-Courant and Monroe is coNP-complete in the setting of approval ballots and rankings. In the latter setting, the result holds for the ℓ_1 and ℓ_∞-Borda misrepresentation functions."

The above result settles the complexity of committee winner verification and recognizes a natural coNP-complete problem, in particular, one that is not merely a complement of a natural NP-complete problem.

We consider the same problem settings for candidate winner verification and by reductions from a vertex cover variant, we obtain the following result (formally stated in Theorem 27):

"**Theorem:** Candidate Winner Verification for Chamberlin-Courant and Monroe is complete for Θ_2^P in the setting of approval ballots and rankings. In the latter setting, the result holds for the ℓ_1 and ℓ_∞-Borda misrepresentation functions."

Given the above hardness results, a natural question is to find structured settings where Problem 1 and 2 can be solved efficiently. We consider the two most well-studies structured settings – single-peaked and single-crossing preferences. Formally, we explain these two structured settings in Section 5.0.2 but at a high-level, under single-peaked preferences, there exists a linear ordering over all candidates and under single-crossing preferences, there exists a linear ordering over the voters which allows for a design of dynamic-programming procedures to obtain efficient algorithms. We summarize our results on these structured preferences as follows (Theorem 29):

"**Theorem:** Committee/Candidate Winner Verification for Chamberlin-Courant are polynomial-time solvable for each of single-peaked and single-crossing preferences. The result holds for the ℓ_1 and ℓ_∞-Borda misrepresentation functions."

1.1.4 Fair Covering of Points in Euclidean Space

In our final problem, we study fairness in max covering of points in Euclidean space. Given a set P of n points in a d-dimensional Euclidean space, our goal is to cover maximum number of points using k unit-radius balls such that the coverage of each color is in *proportion* to its size. We study this problem under the *discreteness* and *bounded ply* constraints, that is, we require the balls used in the covering to be chosen from an input candidate set of unit-radius balls $\mathcal{B}$ (discreteness), and we want any point in the plane to be covered by at most p chosen disks where p is a given constant (bounded-ply). Formally, the input is a set P of n points in $\mathbb{R}^d$ each of which is colored with one of t-colors, a candidate set $\mathcal{B}$ of m unit-radius balls in $\mathbb{R}^d$, budget k of balls to be use, and a number p which is the bound on the ply of the covering. Our goal is to find (approximately) fair covering $\mathcal{C} \subseteq \mathcal{B}$ that covers maximum number of points. We say that $\mathcal{C}$ is a (proportionally) *fair covering* if, let c_i be the number of points of color i covered by $\mathcal{C}$, and let n_i be the total number of points of color i for $i \in \{1, 2, \ldots, t\}$ then,

$$\lfloor \rho_i \cdot c^* \rfloor \leq c_i \leq \lceil \rho_i \cdot c^* \rceil$$

where $c^* = \sum_{i=1}^{t} c_i$ and $\rho_i = n_i/n$ for $i \in \{1, \ldots, t\}$.

Connection to Committee Selection Problems. The fair covering problem stated as above can be formulated as a (fair) committee selection problem under approval preferences in the Euclidean space. That is, the points in P corresponds to the voters and unit-radius balls in $\mathcal{B}$ corresponds to the candidates. Voters give approval preferences based on Euclidean distances, in particular, a voter approves all candidates within a unit-radius ball around it. Given this mapping, the vanilla max covering problem where goal is to cover maximum number of points irrespective fairness constraints corresponds to finding size k committee with max ℓ_1-score committee under approval preferences, and the fair covering problem corresponds to a variant of max ℓ_1-score committee selection under fairness constraints.

We first describe our results for the one-dimensional instances when the points in P are on a line and $\mathcal{B}$ is a set of unit intervals. Following is the summary of our results from Theorem 33 and Theorem 34:

"**Theorem:** For one-dimensional instances, the fair covering problem can be solved polynomial time if the number of colors t is a constant, and the problem is NP-hard when t is large, that is, for $t = \Omega(n)$"

In dimensions $d \geq 2$, the maximum coverage problem is NP-hard even without fairness constraints [19]; therefore, in this case we design an algorithm to find an approximate fair covering.

A covering $\mathcal{C}$ is called ε-fair for some $\varepsilon \in [0, 1]$, if for all $i \in \{1, \ldots, t\}$,

$$(1 - \varepsilon) \cdot \lfloor \rho_i \cdot c^* \rfloor \leq c_i \leq (1 + \varepsilon) \cdot \lceil \rho_i \cdot c^* \rceil$$

We obtain a PTAS (Theorem 36) based on grid shifting technique [17]:

"**Theorem:** There exists a $(1 - \varepsilon)$-approximation algorithm for the (t-color) fair covering problem in $\mathbb{R}^d$ which runs in $n^{\mathcal{O}(t)} m^{\mathcal{O}(1/\varepsilon^d)}$ time."

1.2 Organization of Chapters

We organize the chapters to match the order of the problems described above. In Chapter 2, we study winner determination under Chamberlin-Courant voting rule in Euclidean elections. Recall that, our score is defined by the worst-case representation rank of the voter. We show that the problem is NP-hard and hard to even approximate within any constant factor of the optimal score in dimensions $d \geq 2$. Then we present three polynomial-time approximation schemes, each of which finds a committee with a good minimax score.

Chapters 3 and 4 deals with fault tolerance in committee selection. We introduce and study three – ORP, FTS, and OFTC as described in Section 1.1.2. In Chapter 3, we consider one-dimensional instances when voters and candidates are points on a line and show that all three problems can be solved in polynomial time. We investigate instances in dimensions $d \geq 2$ in Chapter 4. Here, we first show that all three problems are NP-hard and then present constant-factor approximations for all three along with an FPT bicriterion approximation for OFTC.

Next, in Chapters 5 and 6, we consider winner verification problems. We show that Committee Winner Verification is coNP-complete for Chamberlin-Courant and Monroe rules in Chapter 5. We end this chapter with efficient algorithms for Chamberlin-Courant rule on single-peaked and single-crossing preferences in Section 5.4. Chapter 6 deals with Candidate Winner Verification where we show that the problem is θ_2^P-complete for both the rules and is again, polynomial-time solvable for Chamberlin-Courant on restricted preferences.

Finally, in Chapter 7 we study the fair covering problem in Euclidean space and then in Chapter 8 we conclude with some interesting open problems.

1.3 Permissions and Attributions

Most of the research presented in this dissertation have appeared as a conference proceedings. The specific details on chapters is as follows:

1. All results in Chapter 2 is a join work with Subhash Suri and Jie Xue. These results appeared as a paper [20] in proceedings of IJCAI'22 and is available online at https://www.ijcai.org/proceedings/2022/0068.pdf.

2. Results in Chapters 3 and 4 is a joint work with Subhash Suri and Jie Xue, and parts of it have previously appeared as a paper [21] in proceedings of ESA'23 and is available online at https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ESA.2023.95.

3. The contents of Chapters 5 and 6 is a joint work with Neeldhara Misra and Palash Dey. Most results in these two chapters have been appeared previously as a paper [22] in proceedings of IJCAI'20 and is available online at

   https://www.ijcai.org/proceedings/2020/0013.pdf.

4. The contents of Chapter 7 is a joint work with Daniel Lokshtanov, Subhash Suri, and Jie Xue. Most results in this chapter have been appeared previously as a paper [23] in the proceedings of CCCG'20 and is available online at

   https://par.nsf.gov/servlets/purl/10309897page=36.

During my PhD studies, I also worked on other problems that resulted in the following published and in-submissions works:

- Úrsula Hebert-Johnson, Chinmay Sonar, Subhash Suri, and Vaishali Surianarayanan.
  Anonymity-Preserving Space Partitions.
  In the proceedings of ISAAC '21 [24].

- Neeldhara Misra, Chinmay Sonar, P. R. Vaidyanathan, and Rohit Vaish.
  Equitable Division of a Path.
  In the proceedings of COMSOC '21 [25].

- Daniel Lokshtanov, Úrsula Hebert-Johnson, Chinmay Sonar, and Vaishali Suria-narayanan.
  Parameterized Complexity of Kidney Exchange Revisited.
  (In submission to AAAI '24)

- Vaishnavi Himakunthala, Andy Ouyang, Daniel Rose, Ryan He, Alex Mei, Yujie Lu, Chinmay Sonar, Michael Saxon, William Yang Wang.
  Let's Think Frame by Frame: Evaluating Video Chain of Thought with Video Infilling and Prediction.
  In proceedings of EMNLP '23 [26]

- Daniel Rose, Vaishnavi Himakunthala, Andy Ouyang, Ryan He, Alex Mei, Yujie Lu, Michael Saxon, Chinmay Sonar, Diba Mirza, William Yang Wang
  Visual Chain of Thoughts: Bridging Logical Gaps with Multimodal Infillings.
  (In submission to ICLR '24 [27])

Chapter 2

Winner Determination Under the Chamberlin-Courant Rule

In this chapter, we will study the winner determination problem under the Chamberlin-Courant voting rule. We consider the setting when the candidates and voters are placed in a Euclidean space. We will first define the problem formally. As we saw earlier (in Chapter 1), in the committee selection problem we are given an election $E = (\mathcal{C}, V)$ where $\mathcal{C} = \{c_1, c_2, \ldots, c_m\}$ is the set of m candidates and $V = \{v_1, v_2, \ldots, v_n\}$ is the set of n voters. We consider the ordinal preferences where the *preference list* of each voter is a total ordering (ranking) of $\mathcal{C}$, in which the most preferred candidate has rank 1 and the least preferred candidate has rank m. We call an election $E = (\mathcal{C}, V)$ a *d-Euclidean election* if there exists a function $f : \mathcal{C} \cup V \to \mathbb{R}^d$, called a *Euclidean realization* of E, such that for any pair $c_i, c_j \in \mathcal{C}$, a voter v prefers c_i to c_j if and only if $\mathrm{dist}(f(v), f(c_i)) < \mathrm{dist}(f(v), f(c_j))$ where $\mathrm{dist}(\cdot, \cdot)$ denotes the Euclidean distance (in case the candidates are equidistant, we break the ties arbitrarily). We assume that a Euclidean realization is part of the input; the decision problem of whether an election admits an Euclidean realization is computationally hard [28].

We use $\sigma_v(c)$ to denote the rank of candidate c in v's preference list, and we use the Chamberlin-Courant voting rule [11] for evaluating the score of a committee. Recall that under the Chamberlin-Courant voting rule, a voter v's score for a committee $T \subseteq \mathcal{C}$ is the rank of its most preferred committee member $\sigma_v(T) = \min_{c \in T} \sigma_v(c)$, and the *Euclidean minimax committee* problem is to choose a committee of size k that minimizes the maximum score (misrepresentation) of any voter. That is, minimize the following:

$$\sigma(T) = \max_{v \in V} \left(\min_{c \in T} \sigma_v(c) \right),$$

So the optimal committee score is always between 1 and m. The Chamberlin-Courant rule is also known as 1-Borda rule as each candidate is assigned to its single top choice in the committee; in Subsection 2.2.2, we consider a generalization of the Chamberlin-Courant rule which is called the r-Borda rule where each voter is assigned to its top r-choices in the committee and the score of the voter is the sum of the ranks of all candidates it is assigned to.

The winner determination problem is to find a size-k committee $T \subseteq \mathcal{C}$ that minimizes $\sigma(T)$. In the general case (when candidates and voters are not embedded in a Euclidean space), this problem is known to be NP-complete (and hard to approximate within any constant factor) [5, 1], and as a result, an important line of research has been to examine natural settings with *structured* preference spaces [1, 13, 14, 15]. Our work in this chapter studies the setting of elections in the Euclidean space. As discussed in Section 1.1, the number of realizable preference lists in $\mathbb{R}^d$ are polynomially bounded in d (i.e., $\mathcal{O}(m^{d+1})$) compared to all possible $m!$ orderings in the general case; hence, we expect to get much better bounds for Euclidean elections.

Remark: There are good reasons for using *ordinal* preferences even when cardinal distances are implied by an Euclidean embedding. The first is robustness: consider a voter

v and two candidates c, c'. If their distances satisfy $d(v, c) < d(v, c')$, then clearly v prefers c to c', but it seems harder to argue that v's preference varies linearly (or even smoothly) with distance—for instance, would doubling the distance really halve the value to a voter? Another reason is that k-center solutions based on cardinal preferences are highly susceptible to the outlier effect—a few outlying voters may control the minimax value (i.e., radius) of the optimal solution even though all other voters have significantly better solution quality. By contrast, under the ordinal measure the (rank-based) solution seems more equitable because outliers are matched with a highly ranked candidate (irrespective of the distances).

Results and Organization of the Chapter In this chapter we show that a number of interesting and encouraging approximation results are possible for Euclidean preferences. We now give a brief summary of our results.

We first settle the complexity of the problem. To this end, we show that the Euclidean minimax committee problem is NP-hard in every dimension $d \geq 2$; in one dimension, the problem is easy to solve optimally with dynamic programming. The complexity of this problem, also called the ordinal Euclidean k-center problem, was not known and had been an important folklore problem. Our proof shows that the problem also hard to approximate in the worst-case (see Theorem 2), which stands in sharp contrast with the 2-approximability of the *cardinal* k-center problem [8] (please see Section 2.1 for details).

In Section 2.2, we turn to approximation algorithms. Our first result is a polynomial time algorithm to compute a size-k committee with a minimax score of $\mathcal{O}(m/k)$ for any instance in dimension $d = 2$, and score of $\mathcal{O}((m/k) \log k)$ for any instance in dimension $d \geq 3$. Note that this implies an $\mathcal{O}(m/k), O((m/k) \log k)$ approximations in two and more dimensions, respectively. Furthermore, the $\mathcal{O}^*(m/k)$[1] scores are also shown to

[1]The $\mathcal{O}^*(\cdot)$ notation hides the polylog factors

be essentially the best possible in worst-case (see Theorem 4). In Subsection 2.2.2, we also give a similar approximation result to the r-Borda rule. For the special case when $\mathcal{C} \subseteq V$, we show that the optimal has a lower bound of $\Omega(m/k)$ in all instances (see Subsection 2.2.1 for details). Our next approximation uses the bicriterion framework to design a polynomial time algorithm that achieves the *optimal* minimax score $\sigma^\star$ possible for a size-k committee by constructing slightly larger committee, namely, of size $(1+\epsilon)k$ for $d = 2$ and size $O(k \log m)$ for $d \geq 3$. We also show that increasing the committee size by an *additive* constant is not sufficient. We give the details in Subsection 2.3.

Finally, in Section 2.4, our approximation combines ordinal and cardinal features of the problem in a novel way, as follows. Suppose the optimal score of the k committee is $\sigma^\star$, and $d_v^\star$ is the distance of v to its rank $\sigma^\star$ candidate. We define a committee T to be δ-*optimal* if each voter has a representative in T within distance $\delta d_v^\star$. (That is, for each voter the committee contains a candidate whose distance to the voter is almost as good as distance to its σ^* rank candidate.) We show that a δ-optimal committee can be computed in polynomial time for $\delta = 3$, but unless P = NP, there is no polynomial-time algorithm to compute δ-optimal committees for any $\delta < 2$.

2.1 Hardness Results

We begin this section by showing that the Euclidean minimax committee problem is NP-hard in any dimension $d \geq 2$. After that, we extend our proof to show that the problem is even hard to approximate within any sublinear factor of m, where m is the number of candidates in the election.

Our hardness reduction uses the NP-complete problem Planar Monotone 3-SAT (PM-3SAT) [16]. An instance of PM-3SAT consists of a *monotone* 3-CNF formula ϕ where each clause contains either three positive literals or three negative literals, and a

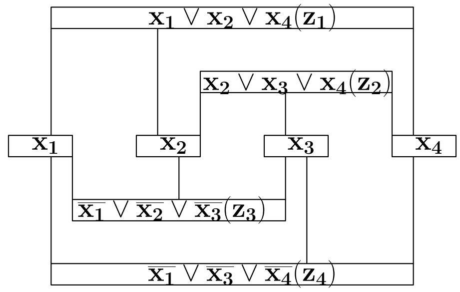


Figure 2.1: Rectangular embedding of the PM-3SAT instance

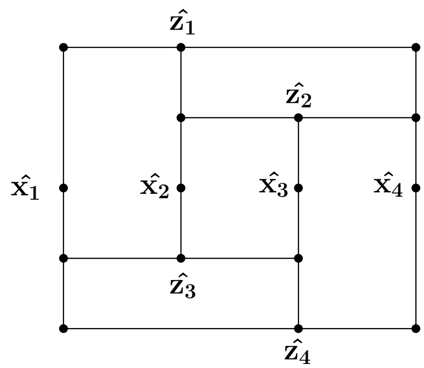


Figure 2.2: Orthogonal embedding of the PM-3SAT instance

special “planar embedding” of the variable-clause incidence graph of ϕ described below. In the embedding, each variable/clause is drawn as a (axis-parallel) rectangle in the plane. The rectangles for the variables are drawn along the x-axis, while the rectangles for positive (resp., negative) clauses lie above (resp., below) the x-axis. All the rectangles are pairwise disjoint. If a clause contains a variable, then there is a vertical segment connecting the clause rectangle and the variable rectangle. Each such vertical segment is disjoint from all the rectangles except the two it connects. We call such an embedding a *rectangular embedding* of ϕ. See Figure 2.1 for an illustration. Given ϕ with a rectangular embedding, the goal of PM-3SAT is to decide if there exists a satisfying assignment for ϕ.

Suppose we are given a PM-3SAT instance consisting of the monotone 3-CNF formula ϕ with a rectangular embedding. Let $X = \{x_1, \ldots, x_n\}$ be the set of variables and

$Z = \{z_1, \ldots, z_m\}$ be the set of clauses of ϕ (each of which consists of three literals). We construct (in polynomial time) a Euclidean minimax committee instance (E, k) in $\mathbb{R}^2$ such that ϕ has a satisfying assignment if and only if for the election E there exists a committee of size k with score at most 4. We begin by modifying the rectangular embedding of ϕ to another form that we call an *orthogonal embedding*.

Orthogonal Embedding First, we collapse all the variable and clause rectangles into horizontal segments: the segments for variables lie on the x-axis and those for positive (resp., negative) clauses lie above (resp., below) the x-axis. By slightly moving the clause segments vertically, we can guarantee that all clause segments have distinct y-coordinates. Consider a variable/clause segment s. There are vertical segments connected to s; we call the intersection points of these segments with s the *connection points*. We then only keep the part of s that is in between the leftmost and rightmost connection points; that is, we truncate the part of s that is to the left (resp., right) of the leftmost (resp., rightmost) connection point. Next, we shrink each variable segment s to a point $\hat{s}$ and also move the vertical segments incident to s accordingly. By doing this, all vertical segments incident to s are merged into a single vertical segment going through $\hat{s}$, whose two endpoints lie on the highest positive s-neighboring clause segment and the lowest negative s-neighboring clause segment; furthermore, this segment hits one endpoint of each of the other s-neighboring clause segments. After shrinking all variable segments, we obtain our orthogonal embedding for ϕ. Figure 1b shows an illustration of the orthogonal embedding. In the orthogonal embedding, each variable corresponds to a point on the x-axis (which we call the *reference point* of the variable) and a vertical segment through the reference point, while each positive (resp., negative) clause corresponds to a horizontal segment above (resp., below) the x-axis; we call the intersection points of these vertical and horizontal segments *connection points*.

The resulting embedding satisfies the following three conditions: *(i)* no vertical segment crosses a horizontal segment; *(ii)* each horizontal segment s intersects exactly three vertical segments which correspond to the three variables contained in the clause corresponding to s; and *(iii)* the endpoints of all segments are *connection points*.

By properties *(ii)* and *(iii)*, there are three connection points on each clause segment, two of which are the left and right endpoints of the segment, and we call the middle one the *reference point* of the clause. We denote by $\hat{x}_1, \ldots, \hat{x}_n$ the reference points of the variables $x_1, \ldots, x_n$ and denote by $\hat{z}_1, \ldots, \hat{z}_m$ the reference points of the clauses $z_1, \ldots, z_m$. By shifting/scaling the segments properly without changing the topological structure of the orthogonal embedding, we can further guarantee that the x-coordinates (resp., y-coordinates) of the vertical (resp., horizontal) segments are distinct *even* integers in the range $\{1, \ldots, 2n\}$ (resp., $\{-2m, \ldots, 2m\}$). Therefore, all the connection points now have integral coordinates (which are even numbers) and the entire embedding is contained in the rectangle $[1, 2n] \times [-2m, 2m]$. Points on the segments of the orthogonal embedding that have integral coordinates partition each segment s into $\ell(s)$ unit-length segments, where $\ell(s)$ is the length of s. We call these unit-length segments the *pieces* of the orthogonal embedding. Let N be the total number of pieces. Clearly, $N = O(nm)$.

Our Euclidean minimax committee instance (E, k) consists of the following set of voters and candidates in the two-dimensional plane. (In fact, in our construction, each point is both a candidate and a voter, namely, $\mathcal{C} = V$. It is easy to modify the construction so that the set of voters is much larger by simply making multiple copies of each voter.)

Variable gadgets. For each variable x_i, we choose four points near the reference point $\hat{x}_i$ as follows. There are two (vertical) pieces incident to $\hat{x}_i$ in the orthogonal embedding, one above $\hat{x}_i$, the other below $\hat{x}_i$. On each of the two pieces, we choose two points with distances 0.01 and 0.02 from $\hat{x}_i$, respectively. We put a candidate and a voter at each of the four chosen points, and call these candidates/voters the x_i-*gadget*. We construct gadgets for all $x_1, \dots, x_n$. The total number of candidates/voters in the variable gadgets is $4n$.

Clause gadgets. The second set of candidates/voters are constructed for the clauses $z_1, \dots, z_m$. For each clause z_i, we put a candidate and a voter at the reference point $\hat{z}_i$, and call this candidate/voter the z_i-*gadget*. The total number of candidates/voters in the clause gadgets is m.

Piece gadgets. The last set of candidates/voters are constructed for connecting the variable gadgets and the clause gadgets. Consider a piece s of the orthogonal embedding, which is a unit-length segment. We distinguish the two endpoints of s as s^- and s^+ as follows. If s is a vertical piece above (resp., below) the x-axis, let s^- be the bottom (resp., top) endpoint of s and s^+ be the top (resp., bottom) endpoint of s. If s is a horizontal piece, then it must belong to the horizontal segment of some clause z_i. If s is to the left (resp., right) of the reference point $\hat{z}_i$, let s^- be the left (resp., right) endpoint of s and s^+ be the right (resp., left) endpoint of s. For every piece s that is *not* adjacent to any clause reference point, we choose four points on s with distances 0.49, 0.8, 0.9, 1 from s^- (i.e., with distances 0.51, 0.2, 0.1, 0 from s^+), respectively. We put a candidate and a voter at each of the four chosen points, and call these the candidates/voters of the s-*gadget*. Note that we do not construct gadgets for the pieces that are adjacent to some clause reference point. Thus, the total number of candidates/voters in the piece gadgets is $4(N - 3m)$, as each clause reference point is adjacent to three pieces.

By combining these three constructed gadgets, we obtain our election $E = (\mathcal{C}, V)$ instance with $4N + 4n - 11m$ candidates and voters. The size of the committee is $k = N + n - 3m$. We now prove that E has a committee of size k with score ≤ 4 iff ϕ is satisfiable.

The "if" part. Suppose ϕ is satisfiable and let $\pi : X \rightarrow \{\mathsf{true}, \mathsf{false}\}$ be an assignment which makes ϕ true. We construct a committee $T \subseteq \mathcal{C}$ of size k as follows. Our committee T contains one candidate in each variable gadget and each piece gadget (this guarantees $|T| = k$ as the total number of variable and piece gadgets is k). Consider a variable x_i. By our construction, the x_i-gadget contains four candidates which have the same x-coordinates as $\hat{x}_i$. If $\pi(x_i) = \mathsf{true}$ (resp., $\pi(x_i) = \mathsf{false}$), we include in T the topmost (resp., bottommost) candidate in the x_i-gadget. Now consider a piece s that is not adjacent to any clause reference point. We first determine a variable as the *associated* variable of s as follows. If s is vertical, then the associated variable of s is just defined as the variable whose vertical segment contains s. If s is horizontal, then s must belong to the horizontal segment of some clause z_j. In this case, we define the associated variable of s as the variable whose vertical segment intersects the left (resp., right) endpoint of the horizontal segment of z_j if s is to the left (resp., right) of the reference point $\hat{z}_j$. Let x_i be the associated variable of s. If $\pi(x_i) = \mathsf{true}$, then we include in T the candidate in the s-gadget that has distance 1 (resp., 0.9) from s^- if s is above (resp., below) the x-axis. Symmetrically, if $\pi(x_i) = \mathsf{false}$, then we include in T the candidate in the s-gadget that has distance 1 (resp., 0.9) from s^- if s is below (resp., above) the x-axis. This finishes the construction of T. The following lemma completes the "if" part of our proof.

Lemma 1 *The score of T in the election E is at most* 4.

Proof: For the reduced Euclidean minimax committee instance $E = (\mathcal{C}, V)$ and the constructed committee $T \subseteq \mathcal{C}$, we will show that, for each voter $v \in V$, $\sigma_v(T) \leq 4$.

Let us begin with the voters in the variable gadgets. For each variable x_i, recall that we place four voters (say v_1, v_2, v_3, v_4) and four candidates (say c_1, c_2, c_3, c_4) near the reference point $\hat{x}_i$. Among the four voters (resp., candidates), v_1, v_2 (resp., c_1, c_2) are at distances 0.01 and 0.02 above $\hat{x}_i$, and v_3, v_4 (resp. c_3, c_4) are at distances 0.01 and 0.02 below $\hat{x}_i$, respectively. We claim that for each of v_1, v_2, v_3, v_4, the closest four candidates to that voter belong to the set $\{c_1, c_2, c_3, c_4\}$. This is because for each of these four voters, any candidate that belongs to its nearest piece gadget is at least at a distance 0.47 from that voter, which is strictly more than the distance from that voter to c_j for all $j \in [4]$, which is bounded by 0.04. Now recall that our constructed committee T either includes the topmost (c_2) or the bottommost (c_4) of the four candidates in the x_i-gadget; hence, the score for each of v_1, v_2, v_3, and v_4 is at most 4. This completes the argument for the voters in all the variable gadgets.

Next, we consider the voters in the piece gadgets. Let s be a piece and let v_1, v_2, v_3, and v_4 (resp., c_1, c_2, c_3, and c_4) be the voters (resp., candidates) at distances $0.49, 0.8, 0.9, 1$ from s^-, respectively. Furthermore, let x_i be the variable associated with s. We only show the proof when s lies above $\hat{x}_i$ (the case when s lies below $\hat{x}_i$ is similar). We first consider the case when $\pi(x_i) = \mathsf{true}$. In this case, T includes c_4. For each of the voters, v_2, v_3 and v_4, their distance from c_4 is at most 0.2. On the other hand, for each of the voters v_2, v_3 and v_4, their distance from a candidate in an adjacent piece gadget is at least at 0.49 (and, their distance from c_1 is at least 0.31). Hence, we conclude that the closest three candidates for v_2, v_3 and v_4 belong to the set $\{c_2, c_3, c_4\}$. Therefore, the score for each these voters is at most 3. Finally, for the voter v_1, $\text{dist}(v_1, c_4) = 0.51$. Hence, v_1's closest four candidates are $\{c_1, c_2, c_3, c'_4\}$ where c'_4 is either the candidate belonging to the preceding piece gadget of s, placed at s^- or it is the candidate placed at a distance 0.02 above the variable reference point $\hat{x}_i$ (we are in the latter case when s is adjacent to $\hat{x}_i$). Therefore, due to the way we construct T, in either case, T includes c'_4. (Note that c'_4 exists even when s^- is a connection point.) Hence, the score of v_1 is at most 4. We now consider the case when $\pi(x_i) = \mathsf{false}$. In this case, T includes the candidate c_3

from s. As argued above, c_3 belongs to the set of closest four candidates for each of the four voters v_1, v_2, v_3, v_4. Hence, the score of committee T for each of the four voters is at most 4.

Finally, we consider the voters from the clause gadget. We only show our argument for an arbitrary positive clause z_i, as the proof for the negative clauses is similar. We need to show that at least one of the four closest candidates to z_i belongs to T. Consider the three pieces s_1, s_2, s_3 adjacent to the clause reference point $\hat{z}_i$. Suppose s_1 is to the left of $\hat{z}_i$, s_2 is to the right of $\hat{z}_i$, and s_3 is below $\hat{z}_i$. By our construction, we have $\hat{z}_i = s_1^+ = s_2^+ = s_3^+$. Since $\hat{z}_i$ is a connection point and all connection points have even coordinates, s_1^-, s_2^-, s_3^- are not connection points. Therefore, there exist pieces s_4, s_5, s_6 such that the right endpoint of s_4 is s_1^-, the left endpoint of s_5 is s_2^-, and the top endpoint of s_6 is s_3^-. We have $s_1^- = s_4^+$, $s_2^- = s_5^+$, and $s_3^- = s_6^+$. In the s_4-gadget, there is a candidate c_4 with distance 1 from s_4^- (and hence located at s_1^-). Similarly, there is a candidate c_5 in the s_5-gadget located at s_2^- and a candidate c_6 in the s_6-gadget located at s_3^-. The candidates c_4, c_5, c_6, together with the candidate in the z_i-gadget (which is located at $\hat{z}_i$), are the four candidates closest to the voter at $\hat{z}_i$, because all pieces except $s_1, \ldots, s_6$ have distances at least 2 from $\hat{z}_i$ by the fact that the x-coordinates (resp., y-coordinates) of the vertical (resp., horizontal) pieces are all even numbers. We claim that T includes at least one of c_4, c_5 or c_6. Recall that π is a valid satisfying assignment. Hence, at least one of the associated variables of s_4, s_5 or s_6 is true under π. Therefore, by the construction of T, for at least one of s_4, s_5 or s_6, T contains the candidate belonging to it which is placed at distance 1 from s_4^-, s_5^- or s_6^-, respectively (i.e., T contains one of c_4, c_5 or c_6). This completes the proof of Lemma 1. ■

The "only if" part. Suppose there exists a size-k committee $T \subseteq \mathcal{C}$ with score at most 4. We use that committee to construct a satisfying assignment $\pi : X \to \{\mathsf{true}, \mathsf{false}\}$. We first note the following property of the committee T.

Lemma 2 *T contains exactly one candidate in each variable gadget and exactly one candidate in each piece gadget.*

Proof: We first show that T contains at least one candidate in each variable gadget. For each variable x_i, recall that we place four voters (say v_1, v_2, v_3, v_4) and four candidates (say c_1, c_2, c_3, c_4) near the reference point $\hat{x}_i$. Among the four candidates (resp., voters), c_1, c_2 (resp., v_1, v_2) are at distances 0.01 and 0.02 above $\hat{x}_i$, and c_3, c_4 (resp. v_3, v_4) are at distances 0.01 and 0.02 below $\hat{x}_i$, respectively. We recall from the proof of Lemma 1 (paragraph 2) that the closest four candidates to each of v_1, v_2, v_3 and v_4 belong to the set $\{c_1, c_2, c_3, c_4\}$. Since the score of committee T is at most 4, then, in particular, $\sigma_{v_i}(T) \leq 4$ for each $i \in [4]$; hence, T contains at least one of c_1, c_2, c_3, c_4. This completes the argument for the variable gadgets. Next, consider a piece gadget s. Let v_1, v_2, v_3, and v_4 (resp., c_1, c_2, c_3, and c_4) be the voters (resp., candidates) placed at distances $0.49, 0.8, 0.9,$ 1 from s^-, respectively. For the voters v_2 and v_3 a candidate from an adjacent variable/piece/clause gadget is at least at a distance 0.78/0.57/1.1, respectively, while the candidates c_1, c_2, c_3, c_4 lie within distance 0.41. Therefore, the closest four candidates for v_2 and v_3 belong to the set $\{c_1, c_2, c_3, c_4\}$. Hence, T includes at least one of c_1, c_2, c_3, c_4 for each piece gadget.

At this stage, we recall that the number of variable gadgets is n, the number of piece gadgets is $N - 3m$, and the committee size is $k = N + n - 3m$. Hence, using a counting argument, we conclude that T contains exactly one candidate from each variable gadget and each piece gadget. ■

We note that the total number of variable and piece gadgets is k. Since $|T| = k$, using Lemma 2, we conclude that T has no budget to contain any candidate in the clause gadgets.

Corollary 1 *T contains no candidate in the clause gadgets.*

Recall that each variable gadget contains four candidates, two of which are above the x-axis while the other two are below the x-axis. Consider a variable x_i. By Lemma 2, T contains exactly one candidate in the x_i-gadget. If that candidate is above (resp., below) the x-axis, we set $\pi(x_i) = \textsf{true}$ (resp., $\pi(x_i) = \textsf{false}$). We show that π is a satisfying assignment of ϕ. It suffices to show that every positive (resp., negative) clause of ϕ contains at least one variable which is mapped to $\textsf{true}$ (resp., $\textsf{false}$) by π. We only consider positive clauses, as the proof for negative clauses is similar. We need the following property of T.

Lemma 3 *Let s be a piece above the x-axis that is not adjacent to any clause reference point, and suppose x_i is the associated variable of s. If T contains the candidate in the s-gadget with distance 1 from s^-, then $\pi(x_i) = \textsf{true}$.*

Proof: Consider a piece s above the x-axis that is not adjacent to any clause reference point, and let x_i be the variable associated with s. First, consider the case when s is not adjacent to the variable reference point $\hat{x}_i$. Let v_1, v_2, v_3, and v_4 (resp., c_1, c_2, c_3, and c_4) be the voters (resp., candidates) placed at distances $0.49, 0.8, 0.9$, and 1 from s^-, respectively. Moreover, let s' be the piece below (resp., to the left of) s when s is a vertical (resp., horizontal) piece. We assume that $c_4 \in T$ (note that $\text{dist}(s^-, c_4) = 1$). We recall from the proof of Lemma 1 (paragraph 3) that the closest four candidates for v_1 are c_1, c_2, c_3, c'_4 where c'_4 is a candidate from the piece s' placed at a distance 1 from s'^-. Using Lemma 2, we know T only includes c_4 from s. Hence, to satisfy $\sigma_{v_1}(T) \leq 4$, T must include the candidate c'_4. Observe that we can repeat the above argument for all pieces below (resp., to the left of) s, which implies that for all pieces s_i below (resp., to the left of) s, T includes the candidate in s_i placed at a distance 1 from s_i^-.

Let x_i be the variable associated with s and $\hat{s}$ be the piece adjacent to the variable reference point $\hat{x}_i$. Moreover, let $\hat{v}_1, \hat{v}_2, \hat{v}_3$, and $\hat{v}_4$ (resp., $\hat{c}_1, \hat{c}_2, \hat{c}_3$, and $\hat{c}_4$) be the voters (resp., candidates) placed at distances $0.49, 0.8, 0.9$, and 1 from $\hat{s}^-$, respectively. The four closest candidates to $\hat{v}_1$ belong to the set $\{\hat{c}_1, \hat{c}_2, \hat{c}_3, c_5\}$ where c_5 is the candidate at a distance 0.02 above $\hat{x}_i$ added corresponding to the x_i-gadget. Using the argument above, we know T only includes $\hat{c}_4$ from the piece $\hat{s}$. Hence, to satisfy $\sigma_{\hat{v}_1}(T) \leq 4$, T must include the candidate c_5. Recall that for an arbitrary variable x_j for $j \in [n]$, if T includes a candidate above the reference point $\hat{x}_j$, we set $x_j = \mathsf{true}$. Since, $c_5 \in T$ and c_5 lies above $\hat{x}_i$, we set $x_i = \mathsf{true}$. This completes the proof of Lemma 3. ■

We will now use Lemma 2, Lemma 3 and Corollary 1 to show that the constructed assignment π satisfies ϕ. We will show this only for positive clauses as the argument for negative clauses is similar.

Let z_i be a positive clause. We want to show that at least one variable of z_i is mapped to true by π. Consider the three pieces s_1, s_2, s_3 adjacent to the reference point $\hat{z}_i$. Suppose s_1 is to the left of $\hat{z}_i$, s_2 is to the right of $\hat{z}_i$, and s_3 is below $\hat{z}_i$. By our construction, we have $\hat{z}_i = s_1^+ = s_2^+ = s_3^+$. Since $\hat{z}_i$ is a connection point and all connection points have even coordinates, s_1^-, s_2^-, s_3^- are not connection points. Therefore, there exist pieces s_4, s_5, s_6 such that the right endpoint of s_4 is s_1^-, the left endpoint of s_5 is s_2^-, and the top endpoint of s_6 is s_3^-. We have $s_1^- = s_4^+$, $s_2^- = s_5^+$, and $s_3^- = s_6^+$. In the s_4-gadget, there is a candidate c_4 with distance 1 from s_4^- (and hence located at s_1^-). Similarly, there is a candidate c_5 in the s_5-gadget located at s_2^- and a candidate c_6 in the s_6-gadget located at s_3^-. The candidates c_4, c_5, c_6, together with the candidate in the z_i-gadget (which is located at $\hat{z}_i$), are the four candidates closest to the voter at $\hat{z}_i$, because all pieces except $s_1, \ldots, s_6$ have distances at least 2 from $\hat{z}_i$ by the fact that the x-coordinates (resp., y-coordinates) of the vertical (resp., horizontal) pieces are all even numbers. Since the score of T is at most 4, T must contain at least one of these four candidates. However, T does not contain the candidate in the z_i-gadget by Corollary 1. Thus, $T \cap \{c_4, c_5, c_6\} \neq \emptyset$. By Lemma 3, this implies that at least one of the associated variables of s_4, s_5, s_6 is true. Note that these associated variables are just the three variables in the clause z_i. Therefore, π makes z_i true. This completes the "only if" part of our proof. As a result, we see that ϕ is satisfiable iff there exists a committee in E of size k whose score is at most 4.

Finally, the reduction can clearly be done in polynomial time, and so we have established the following result.

Theorem 1 *Euclidean minimax committee is* NP-hard *in all dimensions $d \geq 2$. This claim holds even if the voter and candidate sets are identical.*

In fact, our construction also rules out the possibility of a PTAS as it is hard to decide in polynomial time whether the minimum score is ≤ 4 or ≥ 5. By slightly modifying the proof, we can also show that even computing an approximation for Euclidean minimax committee within any sublinear factor of $|\mathcal{C}|$ is hard.

Theorem 2 *For any constant $\epsilon > 0$, it is* NP-hard *to achieve a $|\mathcal{C}|^{1-\epsilon}$-approximation for Euclidean minimax committee in $\mathbb{R}^d$ for any $d \geq 2$.*

Proof: Our proof is a slight modification of the proof of NP-hardness from Theorem 1. We will show that even computing an approximation for Euclidean minimax committee within a sublinear factor of $|\mathcal{C}|$ is hard.

Given a PM-3SAT instance with formula ϕ, we construct the orthogonal embedding and create the clause gadgets as before. Next, we slightly change the piece gadgets as follows. For each piece s that is not adjacent to a clause reference point, we choose 4 points with distances $0.47, 0.8, 0.9, 1$ from s^- (call them s_1, s_2, s_3, s_4), and introduce a voter and a candidate at each of these points. Also, we create an additional set of candidates as follows. Note that s_4 is an endpoint of s. Thus, there can be two or three pieces adjacent to s_4, depending on whether s_4 is a connection point or not. It follows that when s is vertical (resp., horizontal), there is no piece adjacent to s_4 in one of the left or right (resp., top or bottom) directions. Without loss of generality, assume s is vertical and there is no piece adjacent to s_4 on its left. Let s_4' (resp., s_3') be the point to the left of s_4 (resp., s_3) with distance 0.19 from s_4 (resp., s_3). We then place $(nm)^w$ candidates at s_4' (resp., s_3') for a sufficiently large constant w. The candidates/voters at s_1, s_2, s_3, s_4 and the additional candidates at s_3', s_4' form the s-gadget. Finally, recall that in each x_i-gadget, we have candidates/voters at the four points near $\hat{x}_i$ two of which are at distances 0.01 and 0.02 above $\hat{x}_i$ and the other two of which are at distances 0.01 and 0.02 below $\hat{x}_i$. Let p^+ (resp., p^-) be the point at distance 0.02 above (resp., below) $\hat{x}_i$. Let q^+ (resp., q^-) be the point to the left of p^+ (resp., p^-) with distance 0.05 from p^+ (resp., p^-). We place $(nm)^w$ additional candidates at q^+ (resp., q^-). This completes the construction. The desired committee size is again $k = N + n - 3m$.

We first show the following structural lemma for the constructed instance.

Lemma 4 *If ϕ is satisfiable, then there exists a size k committee with score at most 4; otherwise, every size k committee has score at least $(nm)^w$.*

Proof: First, to show that when ϕ is satisfiable, there exists a size-k committee with score at most 4, we refer the reader to the "if" part of the argument of equivalence for the NP-hardness result in Theorem 1 where we construct a committee T. It can be easily verified that the same committee T also has the score at most 4 in the instance constructed for Lemma 4.

Next, we show that if ϕ is unsatisfiable, then every size-k committee has score at least $(nm)^w$. For a contradiction, assume that when ϕ is unsatisfiable, then in the reduced instance, there is a size-k committee T with score strictly less than $(nm)^w$. We will first show the following claim.

Claim 1 *T contains exactly one candidate in each variable gadget and exactly one candidate in each piece gadget. Moreover, T contains no candidate in the clause gadgets.*

Proof: We first show that T contains at least one candidate in each variable gadget.

For each variable x_i, recall that we place four voters (say v_1, v_2, v_3, v_4) and four candidates (say c_1, c_2, c_3, c_4) near the reference point $\hat{x}_i$. Among the four voters (resp., candidates), v_1, v_2 (resp., c_1, c_2) are at distances 0.01 and 0.02 above $\hat{x}_i$, and v_3, v_4 (resp. c_3, c_4) are at distances 0.01 and 0.02 below $\hat{x}_i$, respectively. Moreover, let q^+ (resp., q^-) be the point to the left of v_2 (resp., v_4) at a distance 0.05; we place $(nm)^w$ candidates at q^+ (resp., q^-). Observe that the closest $(nm)^w$ candidates for the voters v_1, v_2, v_3, v_4 belong to the x_i-gadget. This is because any candidate from the nearest piece gadget is at least at a distance 0.4. Hence, T contains at least one candidate from each variable gadget. Next, we consider a piece gadget s. We first recall the construction of s. Let s_1, s_2, s_3, and s_4 be the points placed at distances $0.47, 0.8, 0.9, 1$ from s^-, and let s'_3, s'_4 be the points placed at a distance 0.19 to the left of the points s_3, s_4, respectively. We place a voter and a candidate at each of s_1, s_2, s_3 and s_4, and place $(nm)^w$ candidates at s'_3 and s'_4. Notice that the closest $(nm)^w$ candidates for the voter placed at s_4 include the candidates placed s_4, s_3 and s'_4 (which belong to s). Hence, T contains at least one candidate from each piece gadget.

At this stage, we recall that the number of variable gadgets is n, the number of piece gadgets is $N - 3m$, and the committee size is $k = N + n - 3m$. Hence, using a counting argument, we conclude that T contains exactly one candidate from each variable gadget and each piece gadget. Clearly, T cannot contain any candidate from the clause gadgets. This completes the proof of Claim 1. ■

Next, we strengthen Claim 1 as follows: For each piece gadget s, we show that T contains the candidate placed at either s_3 or s_4. Indeed, consider the voters placed at s_3

and s_4. The closest $(nm)^w$ candidates for the voter at s_3 include the candidates placed at s_1, s_2, s_3, s_4, s_3'. And, for the voter at s_4, it's closest $(nm)^w$ candidates include the candidates placed at s_3, s_4, s_4'. By Claim 1, we know T includes exactly one candidate in s; hence, T must include a candidate placed at either s_3 or s_4 to bound the score for the voters at s_3 and s_4 simultaneously. By a similar argument, for each variable gadget, we can show that T must include c_1, c_2, c_3 or c_4 (in particular, T cannot include any candidates placed at q^+ or q^-). Therefore, we have shown that T does not contain any candidate placed on one of the points with $(nm)^w$ candidates.

We now claim that we can recover a satisfying assignment π for the PM-3SAT instance ϕ. We define π as follows: For each variable x_i, if T includes a candidate from the x_i-gadget placed above the x-axis, then $\pi(x_i) = \mathsf{true}$; otherwise $\pi(x_i) = \mathsf{false}$. We can now use a similar argument to the "only if" part of the argument of equivalence for the NP-hardness result from Theorem 1 to show that π is a satisfying assignment of ϕ (due to the similarity of the arguments, we skip the details). But this is a contradiction, since we started with an unsatisfiable instance ϕ of PM-3SAT. This completes the proof of Lemma 4. ■

Our construction satisfies $|\mathcal{C}| = (4N+4n-11m)+(nm)^w N$, and since $N = O(nm)$ and we can choose any sufficiently large value for w, we can guarantee that $(nm)^w \geq |\mathcal{C}|^{1-\epsilon}$ for any small constant $\epsilon > 0$. This completes the proof of Theorem 2. ■

In the rest of this chapter, we complement the hardness results of the previous sections with nearly-optimal approximation algorithms.

2.2 Approximation using epsilon-nets

Our first algorithm computes in polynomial-time a size-k committee of minimax score $O(m/k)$ for $d = 2$ and $O((m/k)\log k)$ for $d \geq 3$. Our algorithm uses the notion of ϵ-

nets, which are commonly used in computational geometry [29] for solving set cover and hitting set problems. Let us first briefly describe this notion.

Let X be a finite set of points in $\mathbb{R}^d$ and let $\mathcal{R}$ be a set of ranges (subsets of X) in $\mathbb{R}^d$. A subset $A \subseteq X$ is called an ϵ-*net* of $(X, \mathcal{R})$ if A intersects all those ranges in $\mathcal{R}$ that are ϵ-heavy, i.e., they contain at least an ϵ-fraction of the points in X. In other words, A is an ϵ-net for $(X, \mathcal{R})$ if $A \cap R \neq \emptyset$ for any $R \in \mathcal{R}$ with $|R \cap X| \geq \epsilon |X|$. There exists an ϵ-net of size $O(\frac{1}{\epsilon})$ for ranges defined by disks in $\mathbb{R}^2$, and of size $O(\frac{1}{\epsilon} \log \frac{1}{\epsilon})$ for ranges defined by balls in $\mathbb{R}^d$, for any constant dimension $d \geq 3$ [29]. In both cases, ϵ-nets can be computed in polynomial time.

Building on this result, we now present our algorithm.

Theorem 3 *Given a d-Euclidean election $E = (\mathcal{C}, V)$, we can compute in polynomial time a size k committee with minimax score $O(m/k)$ for $d = 2$ and score $O((m/k) \log k)$ for $d \geq 3$, where $m = |\mathcal{C}|$.*

Proof: In order to convey the intuition more clearly, let us first show how to find an $O(k)$-size committee with score at most $\lceil (m/k) \log k \rceil$. Given a d-Euclidean election, let $\mathcal{C}$ be the set of the m candidates with their embedding in $\mathbb{R}^d$. For each voter v, we consider a d-dimensional ball R_v centered at v containing the $\lceil (m/k) \log k \rceil$ closest points of $\mathcal{C}$ to v. Let $\mathcal{R}$ be the set of all these balls. Each ball of $\mathcal{R}$ is ϵ-heavy for $\epsilon = \log k/k$ because it contains an ϵ-fraction of the m candidates. Therefore, in polynomial time we can find an ϵ-net $T \subseteq \mathcal{C}$ for $(\mathcal{C}, \mathcal{R})$ of size $O(\frac{1}{\epsilon} \log \frac{1}{\epsilon}) = O(k)$. By the definition of ϵ-net, T contains at least one point from each R_v, and thus points of T form a committee of size $O(k)$ with minimax score $\lceil (m/k) \log k \rceil$.

To reduce the committee size to k while increasing the score by only a constant factor, we enlarge each ball R_v to include the $\alpha(m/k) \log k$ closest candidates of v, for an appropriate constant α. Each ball is now ϵ'-heavy, for $\epsilon' = \alpha \log k/k$, which guarantees an ϵ'-net $T \subseteq \mathcal{C}$ for $(\mathcal{C}, \mathcal{R})$ of size $O(\frac{1}{\epsilon'} \log \frac{1}{\epsilon'}) = O(k/\alpha)$. With an appropriate choice of α, we can ensure $|T| \leq k$ and achieve the score of $\alpha(m/k) \log k = O((m/k) \log k)$.

When $d \leq 2$, the ϵ-nets of this set system have size $O(\frac{1}{\epsilon})$, and therefore we can construct a committee of size k with score $O(m/k)$. ■

The $O((m/k) \log k)$ and $O(m/k)$ bounds of Theorem 3 are essentially the best possible. In particular, we can construct instances of Euclidean elections in which no size-k committee can achieve the minimax score better than $\Omega(m/k)$.

Theorem 4 *For any $d \geq 1$, there exist Euclidean elections in $\mathbb{R}^d$ such that any committee $T \subseteq \mathcal{C}$ of size k has score $\Omega(m/k)$, where $m = |\mathcal{C}|$.*

Proof: We give an example of a one-dimensional election instance where any size-k committee has score $\Omega(m/k)$ which can then be embedded in higher dimensions.

Consider a set of n voters on the real line at positions in $[n]$, and a set of m candidates at positions $i(n/m)$, for $i \in [m]$, assuming n is a multiple of m. An optimal size-k committee corresponds to a partition of the line into k pieces, each containing $\Theta(n/k)$ voters. Each such group also contains $\Theta(m/k)$ candidates, and only one of them is in the committee. Therefore, in each group, there is at least one voter (e.g., the leftmost or the rightmost) whose score is $\Omega(m/k)$. ■

2.2.1 Lower bound on the optimal score when $\mathcal{C} \subseteq V$

We now consider a special class of election instances *when the candidate set is a subset of the voter set* (namely, $\mathcal{C} \subseteq V$). We note that most representative elections satisfy this condition because each candidate is also a voter.

Notice that Theorem 4 cannot be used to bound the approximation ratio of our algorithm in Theorem 3, because the lower bound is only derived for the hard instances

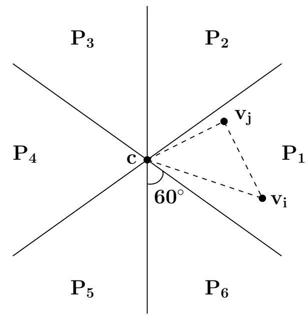


Figure 2.3: Partition of space across the candidate c. Adjacent solid lines form a $60°$ angle at c. The part P_1 contains the voters v_i, v_j.

constructed in the proof. In what follows, we prove that there is a lower bound of $\Omega(m/k)$ on the optimal score for all the instances where $\mathcal{C} \subseteq V$.

Theorem 5 *For a d-Euclidean election $E = (\mathcal{C}, V)$ with $\mathcal{C} \subseteq V$, any size-k committee in E has score $\Omega(m/k)$, where $m = |\mathcal{C}|$.*

Proof: We will only show the proof in $d = 2$ dimensions, since a similar idea works for all constant dimensions $d \geq 3$. Consider a sub-election $E' = (\mathcal{C}', V')$ of E such that $\mathcal{C}' = V' = \mathcal{C}$, i.e., E' only contains those points in E which have both a candidate and a voter placed on them. Hence, the number of candidates and the number of voters in E' is m. We will show that any size-k committee in E' has score $\Omega(m/k)$. Since $V' \subseteq V$ and $\mathcal{C}' = \mathcal{C}$, it implies that the score of any size-k committee in E is also $\Omega(m/k)$. Therefore, for the rest of this proof, we will only work with election E'.

We begin by showing the following structural lemma for the constructed election E'.

Lemma 5 *In $E' = (\mathcal{C}', V')$, each candidate belongs to the set of the closest s candidates for at most $6(s-1)+1$ voters. In other words, a candidate can be one of the top s choices for at most $6(s-1)+1$ voters.*

Proof: The proof is trivial for $s = 1$; hence, we consider the case when $1 < s \leq m$. Consider an arbitrary candidate $c \in \mathcal{C}'$. We equipartition the space into six parts

$P_1, P_2, \ldots, P_6$ using three lines across c (see Figure 2.3). We assume that no candidate or voter lies on any of these three lines (note that this can be ensured by slightly moving the candidates/voters while ensuring that the rankings for each voter does not change).

We claim that c belongs to the set of the closest s candidates for at most s voters in each part P_i for $i \in [6]$. Without loss of generality, we prove our claim for P_1. Let $\hat{P} = \{\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_q\}$ where $q \geq s$ (when $q \leq s$, the proof is trivial) be the points in P_1 which have a voter and a candidate. Note that $\hat{P}$ is sorted according to the distance of it's points from c. Let $\hat{V} = \{\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_q\}$ and $\hat{C} = \{\hat{c}_1, \hat{c}_2, \ldots, \hat{c}_q\}$ be the set of voters and candidates, respectively, such that $\hat{v}_i, \hat{c}_i$ are located at $\hat{p}_i$. We will show that for each pair $i, j \in [q]$ with $j < i$, v_i prefers c_j to c. We need to show that $\text{dist}(v_i, c_j) < \text{dist}(v_i, c)$. Recall that $\text{dist}(v_i, c) > \text{dist}(v_j, c)$. Hence, using the sine rule, we know $\angle cv_jv_i > \angle cv_iv_j$. Furthermore, if $\text{dist}(v_i, c_j) > \text{dist}(v_i, c)$, then using the sine rule, we get $\angle v_icv_j > \angle cv_jv_i > \angle cv_iv_j$. But observe that $\angle v_icv_j < 60°$. This implies that $\angle v_icv_j + \angle cv_jv_i + \angle cv_iv_j < 180°$, which is a contradiction. Hence, $\text{dist}(v_i, c_j) < \text{dist}(v_i, c)$. This completes the proof of Lemma 5. ■

Let $T = \{t_1, t_2, \ldots, t_k\}$ be an optimal committee in E'. We partition the set of voters V' into k parts $V'_1, V'_2, \ldots, V'_k$ such that voters in V'_i have t_i as their most preferred candidate in T for $i \in [k]$. Using an averaging argument, we know that there exists some index $j \in [k]$ such that V'_j contains at least m/k voters. Since all the voters in V'_j are represented by t_j, using Lemma 5, we conclude that there is a voter $v \in V'_j$ such that $\sigma_v(T) \geq \frac{m}{6(k-1)+1}$. This completes the proof of Theorem 5. ■

In light of the above lower bound, it follows that whenever the candidate set is a subset of the voter set, Theorem 3 implies an $O(1)$-approximation of the minimax score for $d = 2$, and an $O(\log k)$-approximation for $d \geq 3$.

Corollary 2 *Given a d-Euclidean election $E = (\mathcal{C}, V)$ with $\mathcal{C} \subseteq V$, we can compute in* *polynomial time a size-k committee with minimax score within a constant factor of the optimal for $d = 2$, and within a factor of $O(\log k)$ for $d \geq 3$.*

2.2.2 The r-Borda Rule

A natural generalization of the Chamberlin Courant rule is the so-called *r-Borda rule* where the score of each voter is determined by its nearest r candidates in the committee for a given $r \leq k$. More specifically, the score of a voter v with respect to a committee is the sum of the ranks of its nearest r candidates in the committee in the preference list of v. The minimax score of a committee T is the maximum over all voter scores. Our goal is to find a committee T of size k that minimizes $\sigma(T)$, where

$$\sigma(T) = \max_{v \in V} \left(\min_{Q \subseteq T, |Q| = r} \left(\sum_{c \in Q} \sigma_v(c) \right) \right).$$

We show that for any election $E = (\mathcal{C}, V)$ in $\mathbb{R}^d$, we can compute a size-k committee with an r-Borda score of $O((r^2 m/k) \log k)$ using a modification of the algorithm in Theorem 3.

Theorem 6 *Given an election in any fixed dimension d, we can find in polynomial time a size-k committee with minimax r-Borda score $O((r^2 m/k) \log k)$. Furthermore, if $d \leq 2$, the score can be further improved to $O(r^2 m/k)$.*

Proof: We only give a high-level idea of our algorithm as the rest of the details are similar to the proof of Theorem 3.

For each voter $v \in V$, we create a ball R_v centered at v that contains $\lceil \alpha(rm/k) \log k \rceil + r$ candidates for a sufficiently large constant α. Let $\mathcal{R} = \{R_v : v \in V\}$ and $\epsilon = \alpha(r/k) \log k$. Our algorithm runs in r rounds. In each round, we first compute an ϵ-net $T_0 \subseteq \mathcal{C}$ for $(\mathcal{C}, \mathcal{R})$ of size $O(\frac{1}{\epsilon} \log \frac{1}{\epsilon})$. Since α is sufficiently large, we have $|T_0| \leq k/r$.

Then we add the candidates in T_0 to the committee T and remove them from $\mathcal{C}$. After r rounds, we obtain our committee T, which is of size at most k. To see that the r-Borda score of T is $O((r^2m/k)\log k)$, we observe that each ball R_v contains at least r candidates in T, which implies that the score of v with respect to T is $r(\lceil \alpha(rm/k)\log k\rceil + r) = O((r^2m/k)\log k)$. Recall that T is the (disjoint) union of r ϵ-nets. If R_v contains at least one candidate in each of the r ϵ-nets, then $|R_v \cap T| \geq r$, and we are done. So suppose R_v does not contain any candidate in the ϵ-net generated in the i-th round. This means R_v is not ϵ-heavy in the i-th round. But R_v contains at least $m/\epsilon + r$ candidates in the original $\mathcal{C}$. Therefore, in the i-th round, at least r candidates in R_v are removed from $\mathcal{C}$, and they are already included in T. It follows that $|R_v \cap T| \geq r$.

In $\mathbb{R}^2$, the score of the committee can be further improved to $O(r^2m/k)$ using the same approach with ϵ-nets of size $O(1/\epsilon)$ for disks. This completes the proof of Theorem 6. ∎

We observe that the bound $O(r^2m/k)$ in the above theorem is tight. In particular, there are instances for which an optimal committee's r-Borda score is $\Omega(r^2m/k)$—this can be verified using the instance described in the proof of Theorem 4—and so this serves as the benchmark score for r-Borda.

2.3 Bicriterion Approximation by Relaxing the Committee Size

The hardness result of Theorem 2 rules out any efficient algorithm for Euclidean election with a good approximation guarantee *but only under the rigid constraint that the committee size is at most k.* In this section, we study the problem in the setting where we are allowed to relax the committee size. Specifically, we ask the following natural question: *Can we efficiently compute a committee of size slightly larger than k*

whose score is close to the optimal score of a size-k committee?

First, we show that if we are allowed to increase the committee size by a (small) *multiplicative* factor, then one can achieve (or improve) the optimal score of a size-k committee.

Theorem 7 *Given a d-Euclidean election, we can compute in polynomial time a committee of size $(1+\epsilon)k$, given any fixed $\epsilon > 0$, when $d = 2$ and of size $O(k \log m)$, where m is the number of candidates, when $d \geq 3$, whose score is smaller than or equal to the score of any size-k committee.*

Proof: We prove the result for a Euclidean election $E = (\mathcal{C}, V)$ in $d = 2$; the proof for higher dimensions is similar. Suppose we know that the optimal size-k committee of E has score $\sigma^\star$. We show how to compute a committee of size $(1+\epsilon)k$ whose score is at most $\sigma^\star$. For each voter $v \in V$, we consider the smallest disk R_v centered at v containing its closest $\sigma^\star$ candidates in $\mathcal{C}$. Let $\mathcal{R} = \{R_v : v \in V\}$. A *hitting set* of the set system $(\mathcal{C}, \mathcal{R})$ is a subset $H \subseteq \mathcal{C}$ such that $H \cap R \neq \emptyset$ for all $R \in \mathcal{R}$. If a committee has score at most $\sigma^\star$, then it must be a hitting set of $(\mathcal{C}, \mathcal{R})$, and since there is a size-k committee with score $\sigma^\star$, the minimum hitting set of $(\mathcal{C}, \mathcal{R})$ has size at most k. By using the PTAS for disk hitting set [30], we can compute a hitting set for $(\mathcal{C}, \mathcal{R})$ of size $(1+\epsilon)k$, which is the desired committee. Since we do not know the value of $\sigma^\star$, we simply try all values from 1 to m, and pick the smallest one for which we have a hitting set of size $(1+\epsilon)k$.

In higher dimensions, we can apply the same approach. The only difference is that we do not have a PTAS for ball hitting set in $\mathbb{R}^d$ for $d \geq 3$. But we can apply the greedy hitting set algorithm to compute a hitting set of size $O(k \log m)$ if a size-k hitting set exists. Therefore, the above algorithm computes a committee of size $O(k \log m)$ whose score is at most the score of any size-k committee. ■

On the other hand, we prove that if we can only increase the committee size by an *additive* constant, we are not able to achieve any good approximation for the minimax score.

Theorem 8 *Let $\alpha, \epsilon > 0$ be constants. Given a Euclidean election $E = (\mathcal{C}, V)$ in $\mathbb{R}^d$ for $d \geq 2$ and a number $k \geq 1$, it is* NP-hard *to compute a committee of size $k + \alpha$ whose score is at most $|\mathcal{C}|^{1-\epsilon} \cdot \sigma^\star$, where $\sigma^\star$ is the minimum score of a size-k committee.*

Proof: Our proof is a minor modification of the proof of Theorem 2; hence, we describe the construction and only give the main idea of the proof of equivalence.

We begin by constructing an election instance $E' = (\mathcal{C}', V')$ as described in the proof of Theorem 2 and do the further modifications as follows: Let $E = (\mathcal{C}, V)$ be an election instance constructed by taking $\alpha + 1$ distinct copies of E', i.e., $\mathcal{C} = \mathcal{C}'_1 \cup \mathcal{C}'_2 \cup \cdots \cup \mathcal{C}'_{\alpha+1}$ and $V = V'_1 \cup V'_2 \cup \cdots \cup V'_{\alpha+1}$. In the Euclidean embedding of the election E, we keep the Euclidean embeddings of the individual copies of E' the same as before but place these $\alpha + 1$ copies far away from each other so that any voter $v \in V'_i$ prefers all candidates in its copy to any candidate in any other copy. Recall that in E', the committee size is $k' = N + n - 3m$ according to the construction in Theorem 2 (where N, n, and m are the total number of pieces, variable, and clauses, in the orthogonal embedding of the PM-3SAT instance, respectively). For the election E we set the committee to $k = (\alpha + 1)k'$. This completes the construction of the reduced instance.

The main idea here is that any committee T for the reduced election instance E can be viewed as the disjoint union of candidates in the committees (T'_i) for each individual election E'_i. This is because a voter $v \in V'_i$ prefers any candidate $c_i \in \mathcal{C}'_i$ to a candidate $c_j \in \mathcal{C}'_j$ for all $i, j \in [\alpha]$ with $i \neq j$. Hence, even if we select a committee of size at most $k + \alpha$ in E (i.e., $|T| \leq k + \alpha$), at least one of the copies of E' (in E) will have a committee of size k' (i.e. there exists T'_i such that $|T'_i| \leq k'$). Without loss of generality, assume E'_1 is one such copy. Using Theorem 2, we know that it is NP-hard to achieve even a $|\mathcal{C}_1'|^{1-\epsilon}$-approximation for Euclidean minimax committee in E_1' for any $\epsilon > 0$. Since we consider minimax Chamberlin-Courant rule, the approximation factor in E is the max over the approximation factors in all copies of E' in E. This implies that a polynomial-time algorithm cannot achieve a $|\mathcal{C}_1'|^{1-\epsilon}$-approximation for any $\epsilon > 0$ in E. Since $|\mathcal{C}| \leq (\alpha+1)|\mathcal{C}_1'|$, where α is a constant, this completes the proof of Theorem 8. ■

2.4 Approximation by combining Cardinal and Ordinal Score

In the previous section, we showed that for any instance we can find a minimax committee of optimal score if we increase the committee size by a small (multiplicative) factor. In this section, we suggest an alternative way to assess the approximation quality *while keeping the committee size k.*

To introduce this criterion, let us consider an election $E = (\mathcal{C}, V)$ and suppose the optimal score of a size-k committee is $\sigma^\star$. In our approximation, we are looking for a size-k committee in which the candidate closest to each voter $v \in V$ has rank not much larger than $\sigma^\star$ in the preference list of v. Our hardness proof shows that in general this is not possible because there may be many candidates at roughly the same distance from v, *but with a large difference in ranks*, and any polynomial time algorithm is bound to end up with a bad minimax score for some v. A natural way to get rid of this pathological situation is to treat two candidates with roughly the same distance from v as if they have similar ranks.

With this motivation, we introduce the following *δ-optimality criterion.* We say a committee $T \subseteq \mathcal{C}$ of size k is *δ-optimal*, for $\delta \geq 1$, if for each voter $v \in V$ the distance from v to its closest candidate in T is at most δ times the distance from v to its rank-$\sigma^\star$

candidate.

We now show how to compute a 3-optimal committee in polynomial time for any d-Euclidean election. Let $E = (\mathcal{C}, V)$ be a Euclidean election and $k \geq 1$ be the desired committee size. For convenience, let us first assume that the optimal score $\sigma^\star$ of a size-k committee of E is known. For each voter $v \in V$, define $d_v^\star$ as the distance from v to the rank-$\sigma^\star$ candidate in the preference list of v. We say a voter v is *satisfied* with a subset $T \subseteq \mathcal{C}$ if there exists a candidate $c \in T$ such that $\text{dist}(c, v) \leq 3d_v^\star$. We denote by $S[T] \subseteq V$ the subset of voters satisfied with T. Then a committee $T \subseteq \mathcal{C}$ (of size k) is 3-optimal if every $v \in V$ is satisfied with T. Our algorithm begins with an empty committee $T = \emptyset$ and iteratively adds new candidates to T using the following three steps until $S[T] = V$:

1. $\hat{v} \leftarrow \arg\min_{v \in V \setminus S[T]} d_v^\star$.
2. $\hat{c} \leftarrow$ a candidate within distance $d_{\hat{v}}^\star$ from $\hat{v}$.
3. $T \leftarrow T \cup \{\hat{c}\}$.

In words, in each iteration, we find the *unsatisfied* voter $\hat{v}$ with the minimum $d_{\hat{v}}^\star$, and then add to T a (arbitrarily chosen) candidate $\hat{c} \in \mathcal{C}$ within distance $d_{\hat{v}}^\star$ from $\hat{v}$. The algorithm terminates when $S[T] = V$, and so all voters are satisfied with T at the end.

We only need to show that $|T| \leq k$; if $|T| < k$, we can always add extra candidates while keeping all voters satisfied. Consider an optimal size-k committee $T_{\text{opt}} = \{c_1, \ldots, c_k\}$, with score $\sigma^\star$, and let $V_i \subseteq V$ be the subset of voters whose closest candidate in T_{opt} is c_i. Thus, $V_1, \ldots, V_k$ form a partition of V. We say two voters $v, v' \in V$ are *separated* if they belong to different V_i's.

Suppose our algorithm terminates in r iterations. We will show that $r \leq k$. Let $\hat{v}_j$ (resp., $\hat{c}_j$) be the voter $\hat{v}$ (resp., the candidate $\hat{c}$) chosen in the j-th iteration, and let

T_j be the committee T at the beginning of the j-th iteration. We claim the following property of our greedy algorithm.

Lemma 6 *The voters $\hat{v}_1, \dots, \hat{v}_r$ are pairwise separated.*

Proof: Let $j, j' \in [r]$ such that $j \neq j'$, and we want to show that $\hat{v}_j$ and $\hat{v}_{j'}$ are separated. Without loss of generality, assume $j < j'$ and let $i \in [k]$ be such that $\hat{v}_j \in V_i$. We claim that $V_i \subseteq S[T_{j+1}]$. Consider a voter $v \in V_i$. If $v \in S[T_j]$, then $v \in S[T_{j+1}]$. So assume $v \in V_i \backslash S[T_j]$. In this case, $d^\star_{\hat{v}_j} \leq d^\star_v$. Since the score of T_{opt} is $\sigma^\star$, we have $\text{dist}(c_i, v) \leq d^\star_v$ and $\text{dist}(c_i, \hat{v}_j) \leq d^\star_{\hat{v}_j} \leq d^\star_v$. Furthermore, by the construction of $\hat{c}_j$, we have $\text{dist}(\hat{c}_j, \hat{v}_j) \leq d^\star_{\hat{v}_j} \leq d^\star_v$. It follows that

$$\text{dist}(\hat{c}_j, v) \leq \text{dist}(\hat{c}_j, \hat{v}_j) + \text{dist}(\hat{v}_j, c_i) + \text{dist}(c_i, v) \leq 3d^\star_v.$$

Therefore, v is satisfied with $T_{j+1} = T_j \cup \{\hat{c}_j\}$, i.e., $v \in S[T_{j+1}]$. Based on this, we can deduce that $\hat{v}_{j'} \notin V_i$, because $\hat{v}_{j'} \notin S[T_{j'}]$ and $V_i \subseteq S[T_{j+1}] \subseteq S[T_{j'}]$. Since $\hat{v}_j \in V_i$ and $\hat{v}_{j'} \notin S[T_{j'}]$, $\hat{v}_j$ and $\hat{v}_{j'}$ are separated. ∎

Thus, the voters $\hat{v}_1, \dots, \hat{v}_r$ belong to different V_i's, which implies that $r \leq k$ and $|T| = r \leq k$, proving the correctness of our algorithm.

Finally, notice that in our algorithm we assumed $\sigma^\star$ is known, but this assumption is easy to get rid of. We can try each possible value from 1 to $m = |\mathcal{C}|$, and choose the smallest number $\sigma^\star \in [m]$ for which the algorithm returns a committee of size at most k. Thus, we proved the following.

Theorem 9 *Given a Euclidean election $E = (\mathcal{C}, V)$ in any dimension and $k \geq 1$, one can compute a 3-optimal committee of size k in polynomial time.*

Complementary to the above algorithmic result, we can also show the following hardness result.

Theorem 10 *For any $\delta < 2$, unless P=NP, there is no polynomial time algorithm to compute a δ-optimal committee of size k for a given Euclidean election in $\mathbb{R}^d$ for $d \geq 2$.*

Proof: We will show that if there is a polynomial time algorithm which computes a δ-optimal committee for $\delta < 2$, then it can be used to design a polynomial-time procedure to distinguish between satisfiable and unsatisfiable instances of PM-3SAT.

First, we reduce a PM-3SAT instance ϕ to a d-dimensional Euclidean election. Our construction is a slight modification of the one used in Theorem 1. In particular, for each variable x_i, let s_i be the piece adjacent to the reference point $\hat{x}_i$ placed above it. We modify s_i as follows: First, we remove all points (along with the candidates/voters placed on them) from s_i. Next, we move $\hat{x}_i$ up to a distance 0.5 above s_i^- so that $\hat{x}_i$ lies at the midpoint of s_i, i.e., the x-axis now passes through the midpoint of s_i. Furthermore, we change the variable gadget to only have three points: one at the reference point $\hat{x}_i$, and one above and one below it at a distance of $1/6$. We put a candidate and a voter at each of these three points. The total number of candidates/voters in the variable gadgets is $3n$. Moreover, for each clause gadget z_i, we place only a voter at the clause reference point $\hat{z}_i$. Overall, the clause gadgets contain m voters (and no candidates). Finally, we change each nonempty piece gadget s to only contain three points at distances $1/6, 1/2, 5/6$ from s^-, and we place a candidate and a voter at each of these three points. The total number of candidates/voters in the piece gadgets is $3(N - 3m - n)$ where N is the total number of pieces in the orthogonal embedding of the PM-3SAT instance. Recall that for each of the m clauses, the three pieces adjacent to the clause reference point $\hat{z}_i$ are empty, and for each variable x_i, there is a piece (s_i) which only contains points corresponding to a variable gadget.

Overall, we obtain an election $E = (\mathcal{C}, V)$ that consists of $3N - 8m$ voters and $3N - 9m$ candidates. We now set the desired committee size to be $k = N - 3m$. Using similar arguments to the argument of equivalence for the reduction in Theorem 1, we can show that *E has a committee of size k with score at most 3 iff the PM-3SAT instance is satisfiable.* Next, to show that it is unlikely to have an efficient algorithm to compute δ-optimal committees for any $\delta < 2$, we now state the main structural property of our construction.

Observation 1 *In the constructed election $E = (\mathcal{C}, V)$, for all voters $v \in V$, their closest three candidates are within the distance $1/3$, and the distance to the fourth ranked candidate is $2/3$.*

Let $\delta < 2$ be a constant and let P be a polynomial-time algorithm which computes a δ-optimal committee. We will show that P can distinguish satisfiable and unsatisfiable instances of PM-3SAT in polynomial time. Consider a reduced election instance $E = (\mathcal{C}, V)$ constructed from a PM-3SAT instance ϕ, and let T be the δ-optimal committee returned by P. For a committee T, we compute the distance d_{max} which is the maximum distance of a voter $v \in V$ from its most preferred candidate in T, i.e. $d_{max} = \max_{v \in V}(\min_{c \in T} \operatorname{dist}(v, c))$. Clearly, given a committee T, d_{max} can be computed in a polynomial time.

Lemma 7 *The instance ϕ is satisfiable iff $d_v < 2/3$.*

Proof: We first show that if ϕ is a satisfiable PM-3SAT instance then $d_{max} < 2/3$. Let $E = (\mathcal{C}, V)$ be the election constructed using ϕ. Recall that if ϕ is satisfiable, then $\sigma^* \leq 3$ in the reduced instance E (i.e., the optimal rank in E is at most 3). We now use Observation 1 to conclude that $d^*_v \leq 1/3$ for all voters $v \in V$. Hence, in a δ-optimal committee, each voter has its representative within distance strictly less than $2/3$, i.e., $d_{max} < 2/3$. We now turn to the other direction and show that if $d_{max} < 2/3$ then ϕ is satisfiable. We use Observation 1 to conclude that each voter $v \in V$ is represented by one of its closest three candidates, i.e., $\sigma^* \leq 3$. Therefore, ϕ is satisfiable. This completes the proof for Lemma 7. ■

Hence, algorithm P combined with our reduction gives a polynomial-time procedure to distinguish satisfiable and unsatisfiable instances of PM-3SAT which is impossible, unless P=NP. This completes the proof of Theorem 10. ■

2.5 Bibliographic notes

For a general introduction to multi-winner elections, we refer the reader to the works of [31, 32, 33]. The work of [34] studies the computational complexity and axiomatic properties of various egalitarian committee scoring rules under the general preferences. For computing a winning committee under the Chamberlin-Courant rule, polynomial-time algorithms are known only for restricted preferences such as single-peaked, single-crossing, 1D Euclidean, etc. [1, 14, 15]. Very little is known about the more general d-dimensional Euclidean setting considered in this chapter with the exception of a work of [35], which shows NP-hardness for the *approval set* voting rule for the utilitarian objective in 2-dimensional Euclidean elections.

Constant factor approximations are often easier to achieve under the *utilitarian* objective. For instance, [36] present several nearly-optimal approximation bounds; [37] presents a constant factor approximation for minimizing the weighted sum of ranks of the winning candidates; and [38] presents a constant factor approximation for minimizing the sum when $\sigma_v(c)$ is an arbitrary cardinal value. In contrast, for the minimax objective, mostly inapproximability results are known, and only under general preferences [37]. The work in this chapter is the first attempt to study ordinal preferences for the minimax objective under restricted (geometric) preferences.

Chapter 3

Fault-Tolerant Committee Selection (FTCS)

In this chapter we consider the computational complexity of adding fault tolerance into spatial voting (also known as Euclidean committee selection problem). Recall that in Euclidean committee selection problem, the voters and the candidates are both modeled as points in some d-dimensional space, where each dimension represents an independent *policy issue* that is important for the election, and each voter's preference among the candidates is implicitly encoded by a distance function (we refer the reader to Chapter 2 for a formal definition). For example, in the simplest 1-dimensional setting, voters and candidates are points on a line indicating their real-valued preference on a single issue.

Several aspects of Euclidean committee selection such as axiomatic properties, fairness, winning committee determination have been studied in the past [39, 40, 41] but the work in this chapter is first to consider fault-tolerance, that is, how robust a chosen committee is against the possibility that some of the winning members may default. Committee selection problems model a number of applications in the social sciences and in computer science where such defaults are not uncommon, such as democratic elections, staff hiring, choosing public projects, locations of public facilities, jury selection, cache management, etc. [5, 4, 7, 1, 2, 6, 3]. In this chapter, we are particularly interested in designing algorithms to address questions of the following kind: If some of the winning members default, how badly does this affect the overall score of the committee? Or, how much does the committee score suffer if a *worst-case* subset of size f defaults? Finally, can we *proactively* choose a committee in such a way that it can tolerate up to f faults with the minimum possible score degradation? We begin by formalizing these problems more precisely and then describing our results.

3.0.1 Problem Statement

Let $E = (C, V)$ be a Euclidean election. In this chapter, we will use the Euclidean distance of a voter to the closest committee member to measure the quality of the committee. Formally, the score of a committee $T \subseteq C$ for v is defined as $\sigma(v, T) = \min_{c \in T} d(v, c)$, and the score of T as $\sigma(T) = \max_{v \in V} \sigma(v, T)$. Note that the above scoring function is same as the Chamberlin-Courant voting rule or the k-center objective [12].

The fault tolerance of a committee is parameterized by a positive integer f, which is the upper bound on the number of candidates that can fail.[1] Throughout the Chapter, we use the notation J to denote a failing set of candidates. We are allowed to replace the failing members of J with any set of at most $|T \cap J|$ candidates from $C \setminus J$. We often denote this set of *replacement* candidates by R. However, *we must keep all the non-failing members of T in the committee* — that is, the replacement committee is the set $(T \setminus J) \cup R$ — and throughout the chapter our goal is to optimize this committee's score, namely $\sigma((T \setminus J) \cup R)$.

[1] In our chapter, we will allow any subset of size f from C to fail, so the faults can also include candidates not in the selected committee T. This only makes the problem harder because the adversary can always limit the faults to T, and elimination of candidates from $C \setminus T$ makes finding replacements for failing committee members more difficult.

We consider the following three versions of fault-tolerant committee selection, presented in increasing order of complexity. The first problem is the simplest: given a committee and a failing set, find the best replacement committee.

> **Optimal Replacement Problem** (ORP)
>
> **Input:** An election $E = (C, V)$, a committee $T \subseteq C$ and a failing set $J \subseteq C$.
>
> **Goal:** Find a replacement set $R \subseteq C \setminus J$ of size at most $|T \cap J|$ minimizing $\sigma((T \setminus J) \cup R)$.

Our second problem is to quantify the fault tolerance of a given committee T over worst-case faults. That is, what is the largest score of T's replacement when a worst-case subset of f faults occur? We introduce the following notation as T's measure of j-fault-tolerance, for any $0 \leq j \leq f$: $\sigma_j(T) = \max_{J \subseteq C \text{ s.t. } |J| \leq j} \sigma((T \setminus J) \cup R)$, where R is an optimal replacement set with size at most $|T \cap J|$. We want to compute $\sigma_f(T)$. Occasionally, we also use the notation $\sigma_0(T)$ for the no-fault score of T, namely $\sigma(T)$.

> **Fault-Tolerance Score** (FTS)
>
> **Input:** An election $E = (C, V)$, a committee $T \subseteq C$ and a fault-tolerance parameter f.
>
> **Goal:** Compute $\sigma_f(T)$.

Our third and final problem is to compute a committee with optimal fault-tolerance score.

> **Optimal Fault-Tolerant Committee** (OFTC)
>
> **Input:** An election $E = (C, V)$, a committee size k and a fault-tolerance parameter f.
>
> **Goal:** Find $T \subseteq C$ of size at most k minimizing $\sigma_f(T)$.

3.0.2 Results and Organization of the Chapter

In this chapter, we only consider the one-dimensional Euclidean instances. (In Chapter 4, we study fault-tolerant committee selection problems in multidimensions.) In one dimension, the candidates and voters are points on a line. Surprisingly (unlike non-fault-tolerant versions), fault-tolerant committee problems are nontrivial even in one dimension. In particular, while the Optimal Replacement Problem (ORP) is easily solved by a simple greedy algorithm (see Section 3.1 for details), the other two problems, Fault-Tolerance Score (FTS) and Optimal Fault-Tolerant Committee (OFTC), do not appear to be easy. Our main result in one dimension is the design of efficient dynamic-programming-based algorithms for these two problems. Along the way, we solve a *fault-tolerant* Hitting Set problem for points and unit intervals, which may be of independent interest. Section 3.2 describes our most technical results. In this section, we solve the aforementioned fault-tolerant hitting set problem by deriving a structural observation about hitting sets of unit intervals (Lemma 9), and we use this result to design a dynamic-programming based algorithm which is used solve the FTS problem. In Section 3.3, we use the algorithm from Section 3.2 and additionally design a greedy procedure to solve OFTC.

3.1 Optimal Replacement Problem

In the Optimal Replacement Problem (ORP), we are given a committee $T \subseteq C$ and a failing set $J \subseteq C$, and we must find a replacement set R minimizing the score $\sigma((T \setminus J) \cup R)$, where $|R| \leq |T \cap J|$. Since this score is always the distance between some voter-candidate pair, it suffices to solve the following decision problem: Is there a replacement set with score at most r? We can then try all possible $O(nm)$ distances to find the smallest feasible replacement score.

This decision problem is equivalent to the following hitting set problem: for each voter $v \in V$, let I_v be the interval of length $2r$ centered at v, and let $\mathcal{I} = \{I_v : v \in V\}$ be the set of these n (voter) intervals. A subset of candidates is a hitting set for $\mathcal{I}$ if each interval contains at least one of the candidates. In our problem, we are given a hitting set T and a failing subset of candidates J, and we must find the minimum-size replacement hitting set. Such a replacement is easily found using the standard greedy algorithm, as follows. We first remove all of the intervals from $\mathcal{I}$ that are already hit by a candidate in $T \setminus J$, and we also remove all the failing candidates J from C. For the leftmost remaining interval, we then choose the rightmost candidate c contained in it, add it to R, delete all intervals hit by c, and iterate until all remaining intervals are hit. If we ever encounter an interval containing no candidate, or if the size of the replacement set is larger than $|T \setminus J|$, the answer to the decision problem is no. Otherwise, the solution is R. The greedy algorithm is easily implemented to run in time $\mathcal{O}((m+n)\log(m+n))$. To find the optimal replacement set, we can do a binary search over $O(nm)$ values of r and find the smallest r for which $|(T \setminus J) \cup R| \leq k$.

Theorem 11 *The Optimal Replacement Problem can be solved in time $\mathcal{O}((m+n)\log^2(m+n))$ for one-dimensional Euclidean elections.*

3.2 Fault-Tolerance Score

We now come to the more difficult problem of computing the fault-tolerance score $\sigma_f(T)$ of a committee T in one dimension, which is the worst case over all possible failing sets of T. Once again it suffices to solve the following decision problem: given a size-k committee T and a real number r, can we find a replacement with score at most r for *every* failing subset of size f? Using our hitting set formulation, $\sigma_f(T) \leq r$ if and only if T is an *f-tolerant hitting set* of $\mathcal{I}$, that is, for any failing set $J \subseteq C$ of size at most f, there exists a replacement set $R \subseteq C \setminus J$ such that $|(T \setminus J) \cup R| \leq |T|$ and $(T \setminus J) \cup R$ hits $\mathcal{I}$. (Recall that each member of $\mathcal{I}$ is an interval of length $2r$ centered at one of the voter positions.) We can then compute the fault-tolerance score of T by trying each of the $O(nm)$ voter-candidate distances to find the smallest r for which this decision problem has a positive answer.

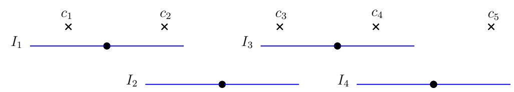


Figure 3.1: The figure shows an interval hitting set instance with four intervals and five points. The set $\{c_2, c_4\}$ is a feasible hitting set. For $X = \{c_2, c_3, c_5\}$, the intervals I_1, I_3, I_4 are X-disjoint.

We solve this fault-tolerant hitting set decision problem by observing that the size of a *smallest* hitting set equals the size of a *maximum* independent set, defined with respect to candidate points and voter intervals in the following way. Suppose the intervals of $\mathcal{I} = \{I_1, \ldots, I_n\}$ are sorted left to right. First, we can assume without loss of generality that $|I_i \cap C| > f$ for all $i \in [n]$, since otherwise there is no f-tolerant hitting set for $\mathcal{I}$. Given a set of points X in $\mathbb{R}$, we say that a set of intervals is X-*disjoint* if each point in X is contained in at most one interval. (That is, X-disjoint intervals can be thought of as *independent* in that they contain disjoint sets of points in X). The following claim is easy to prove.

Lemma 8 *Given a set of points X and a set of intervals $\mathcal{J}$ on the real line, the size of a minimum hitting set $X' \subseteq X$ of $\mathcal{J}$ equals the maximum size of an X-disjoint subset of $\mathcal{J}$.*

Thus, if $T \subseteq C$ is an f-tolerant hitting set for $\mathcal{I}$, then for any failing set $J \subseteq C$, the size of any $(C \setminus J)$-disjoint subset of $\mathcal{I}$ is at most $|T|$. One should note that the size of the maximum $(C \setminus J)$-disjoint subset in $\mathcal{I}$ is a monotonically increasing function of $|J|$ — as more candidates fail, more intervals can become disjoint. Our goal is to find the maximum size of such a disjoint interval family over all possible failure sets J of size at most f. We will do this using dynamic programming, by combining solutions of subproblems, where each subproblem corresponds to an index range $[i, j]$, over the set of candidate points $c_1, \ldots, c_m$. Assuming that the candidate points $C = \{c_1, \ldots, c_m\}$ are ordered from left to right, our subproblems are defined as follows, for $1 \leq i \leq j \leq m$:

- $C_{i,j} = \{c_i, \ldots, c_j\}$ is the set of candidates in the range $[c_i, c_j]$.
- $\mathcal{I}_{i,j} = \{I \in \mathcal{I} : I \cap C \subseteq C_{i,j}\}$ is the set of intervals that only contain points from $C_{i,j}$.
- For any $J \subseteq C_{i,j}$, $\delta_{i,j}(J)$ is the maximum size of a $(C_{i,j} \backslash J)$-disjoint subset of $\mathcal{I}_{i,j}$.
- The subproblems we want to solve are the values $\delta_{i,j}(f) = \max_{J \subseteq C_{i,j}, |J| \leq f} \delta_{i,j}(J)$.

The key technical lemma of this section is the following claim.

Lemma 9 *$T \subseteq C$ is an f-tolerant hitting set of $\mathcal{I}$ if and only if $|T \cap C_{i,j}| \geq \delta_{i,j}(f)$, for all $1 \leq i \leq j \leq m$.*

Proof: We first show the "if" part of the lemma. Assume $|T \cap C_{i,j}| \geq \delta_{i,j}(f)$ for all $i, j \in [m]$ with $i \leq j$. To see that T is an f-tolerant hitting set of $\mathcal{I}$, consider a failing set $J \subseteq C$ of size at most f. We have to show the existence of a replacement set $R \subseteq C \backslash J$ such that $|(T \backslash J) \cup R| \leq |T|$ and $(T \backslash J) \cup R$ is a hitting set of $\mathcal{I}$. We write $T \backslash J = \{c_{i_1}, \ldots, c_{i_p}\}$, where $i_1 < \cdots < i_p$. For convenience, set $i_0 = 0$ and $i_{p+1} = m + 1$. By our assumption, every interval $I \in \mathcal{I}$ is hit by some point in C. Thus, either I is hit by $T \backslash J$ or I belongs to $\mathcal{I}_{i,j}$ where $i = i_{t-1} + 1$ and $j = i_t - 1$ for some index $t \in [p+1]$. Now consider an index $t \in [p+1]$. We write $T_t = T \cap C_{i,j}$ and define $R_t \subseteq C_{i,j} \backslash J$ as a minimum hitting set of $\mathcal{I}_{i,j}$. By Lemma 8, the size of R_t is equal to the maximum size of a $(C_{i,j} \backslash J)$-disjoint subset of $\mathcal{I}_{i,j}$, which is nothing but $\delta_{i,j}(J \cap C_{i,j})$. Also, by assumption, we have $|T_t| = |T \cap C_{i,j}| \geq \delta_{i,j}(f) \geq \delta_{i,j}(J \cap C_{i,j})$. Therefore, $|R_t| \leq |T_t|$. Finally, we define $R = \bigcup_{t=1}^{p+1} R_t$. Clearly, $(T \backslash J) \cup R$ hits $\mathcal{I}$. So it suffices to show that $|(T \backslash J) \cup R| \leq |T|$. Since $|R_t| \leq |T_t|$ for all $t \in [p+1]$, we have

$$|(T \backslash J) \cup R| = |T \backslash J| + \sum_{t=1}^{p+1} |R_t| \leq |T \backslash J| + \sum_{t=1}^{p+1} |T_t| = |T|,$$

which completes the proof of the "if" part.

Next, we prove the "only if" part of the lemma. Assume $T \subseteq C$ is an f-tolerant hitting set of $\mathcal{I}$. Consider two indices $i, j \in [m]$ with $i \leq j$. To show $|T \cap C_{i,j}| \geq \delta_{i,j}(f)$, it suffices to show that $|T \cap C_{i,j}| \geq \delta_{i,j}(J)$ for all $J \subseteq C_{i,j}$ with $|J| \leq f$. Since T is an f-tolerant hitting set of $\mathcal{I}$, there exists $R \subseteq C \backslash J$ such that $|(T \backslash J) \cup R| \leq |T|$ and $(T \backslash J) \cup R$ is a hitting set of $\mathcal{I}$. For brevity, let $T' = (T \backslash J) \cup R$. By definition, the intervals in $\mathcal{I}_{i,j}$ can only be hit by the points in $C_{i,j}$. Thus, $T' \cap C_{i,j}$ is a hitting set of $\mathcal{I}_{i,j}$. As $T' \cap C_{i,j} \subseteq C_{i,j} \backslash J$, by Lemma 8, the size of $T' \cap C_{i,j}$ is at least the maximum size of a $(C_{i,j} \backslash J)$-disjoint subset of $\mathcal{I}_{i,j}$, i.e., $|T' \cap C_{i,j}| \geq \delta_{i,j}(J)$. Furthermore, because $J \subseteq C_{i,j}$, we have $(T \backslash J) \backslash C_{i,j} = T \backslash C_{i,j}$. It follows that $T \backslash C_{i,j} \subseteq T' \backslash C_{i,j}$ and thus $|T \backslash C_{i,j}| \leq |T' \backslash C_{i,j}|$. For a committee T, we can partition T into two parts: the part containing candidates in $C_{i,j}$ and the part containing candidates outside of $C_{i,j}$. Hence, $|T| = |T \cap C_{i,j}| + |T \backslash C_{i,j}|$ and $|T'| = |T' \cap C_{i,j}| + |T' \backslash C_{i,j}|$. Because $|T'| \leq |T|$ and $|T' \backslash C_{i,j}| \geq |T \backslash C_{i,j}|$, we have $|T' \cap C_{i,j}| \leq |T \cap C_{i,j}|$. Therefore, $|T \cap C_{i,j}| \geq \delta_{i,j}(J)$. This completes the proof of Lemma 9. ■

In order to decide if $\sigma_f(T) \leq r$, therefore, we just have to compute $\delta_{i,j}(f)$, for all i, j, and check the condition $|T \cap C_{i,j}| \geq \delta_{i,j}(f)$. We now show how to do that efficiently.

Efficiently Computing $\delta_{i,j}(f)$ For ease of presentation, we show how to compute $\delta_{1,m}(f)$; computing other $\delta_{i,j}(f)$ is similar. We have $C_{1,m} = C$, $\mathcal{I}_{1,m} = \mathcal{I}$, and $\delta_{1,m}(f)$ is size of the largest subset of $\mathcal{I}$ that is $(C \backslash J)$-disjoint for any failing set $J \subseteq C$ with $|J| \leq f$. The intervals of $\mathcal{I} = \{I_1, I_2, \ldots, I_n\}$ are in the left to right sorted order and, for each $i \in [n]$, let $C(I_i) = C \cap I_i$ be the set of points in C that hits I_i. Define $\Gamma[i][j]$ as the maximum size of an $(C \setminus X)$-disjoint subset $\mathcal{J} \subseteq \{I_1, \ldots, I_i\}$ such that $X \subseteq C$ and $|X| \leq j$.

Lemma 10 *We have the following recurrence*

$$\Gamma[i][j] \;=\; \max \left\{ \begin{array}{l} \Gamma[i-1][j], \\ \max\limits_{0 \le i' \le i} 1 + \Gamma[i'][j - |C(I_i) \cap C(I_{i'})|] \end{array} \right\}$$

Clearly, $\delta_{1,m}(f) = \Gamma[n][f]$. The base case for our dynamic program is $\Gamma[0, j] = 0$ for all $j \in [f]$ and $\Gamma[i][j] = -\infty$ for $j < 0$ and all $i \in [n]$. Our dynamic program runs in time $\mathcal{O}(n^2 m f)$. In the same way, we can compute the values of $\delta_{i,j}(f)$ for all $i, j \in [m]$ with $i \le j$.

Lemma 11 $\delta_{i,j}(f)$*, for all* $1 \le i \le j \le m$*, can be computed in time* $\mathcal{O}(n^2 m^3 f)$*.*

Given a hitting set $T \subseteq C$ and the values $\delta_{i,j}(f)$, we can verify the condition in Lemma 9 in time $\mathcal{O}(m^3)$. We can then use binary search to find the smallest value of r for which T is an f-tolerant hitting set. This establishes the following result.

Theorem 12 *The fault-tolerance score of a* 1*-dimensional committee* T *can be computed in time* $\mathcal{O}(n^2 m^3 f \log(nm))$*.*

3.3 Optimal Fault-Tolerance Committee

We now address the problem of *designing* a fault-tolerant committee: select a committee T of size k whose fault-tolerance score $\sigma_f(T)$ is minimized. Thus, our goal is *not* to optimize the fault-free score of T, namely $\sigma_0(T)$, but rather the score that the best replacement will have after a worst-case set of f faults in T, namely $\sigma_f(T)$. Following the earlier approach, we again focus on the decision question: given some $r \ge 0$, is there a committee of size k with $\sigma_f(T) \le r$? For a given value of r, we construct our hitting set instance with candidate-points and voter-intervals, and compute a minimum-sized f-tolerant hitting set $T \subseteq C$ as follows:

1. Compute the value of $\delta_{i,j}(f)$, for all $1 \leq i \leq j \leq m$.

2. Compute a minimum subset $T \subseteq C$ satisfying $|T \cap C_{i,j}| \geq \delta_{i,j}(f)$, for all $1 \leq i \leq j \leq m$.

3. If $|T| \leq k$, we have a solution; otherwise, the answer to the decision problem is no.

Step (1) is implemented using the dynamic program of the previous section, and so it suffices to explain how to implement step (2). We assume without loss of generality that $|C_{i,j}| \geq \delta_{i,j}(f)$ for all i, j, because otherwise there is no solution. We compute a set T using the following greedy algorithm.

- Initialize $T = \emptyset$.

- For each c_k for $k \in [m]$, if there exists $i, j \in [m]$ with $i \leq k \leq j \leq m$ such that $\delta_{i,j}(f) \geq |T \cap C_{i,j}| + (j - k + 1)$, then add c_k to T.

The algorithm runs in time $\mathcal{O}(m^3)$. To prove correctness, we first claim the following.

Lemma 12 *$|T \cap C_{i,j}| \geq \delta_{i,j}(f)$, for all $1 \leq i \leq j \leq m$.*

Proof: Suppose not, so we have $|T \cap C_{i,j}| < \delta_{i,j}(f)$, for some $i \leq j$. We recall that for any interval $I_i \in \mathcal{I}$, $|I_i \cap C| > f$. Therefore, for any failing set J, $C_{i,j} \setminus J$ is a hitting set of $\mathcal{I}_{i,j}$, and $|C_{i,j}| \geq \delta_{i,j}(f)$. This implies that there exists some point among $c_i, \ldots, c_j$ that is not in T. Let $k \in \{i, \ldots, j\}$ be the largest index such that $c_k \notin T$. For convenience, we use T' to denote the set T in the iteration of our algorithm that considers c_k. Note that $T \cap C_{i,j} = (T' \cap C_{i,j}) \cup \{c_{k+1}, \ldots, c_j\}$ and $(T' \cap C_{i,j}) \cap \{c_{k+1}, \ldots, c_j\} = \emptyset$. Therefore, $|T' \cap C_{i,j}| = |T \cap C_{i,j}| - (j-k) < \delta_{i,j}(f) - (j-k)$. This implies $|T' \cap C_{i,j}| + (j-k+1) \leq \delta_{i,j}(f)$. By our algorithm, in this case we should include c_k in T, which contradicts the fact that $c_k \notin T$. ■

We now argue that T has the minimum size among all subsets of C satisfying the property of Lemma 12. Let opt be the minimum size of a subset of C satisfying the desired property. We write $T = \{c_{k_1}, \ldots, c_{k_r}\}$, where $k_1 < \cdots < k_r$.

Lemma 13 *For any $t \in [r]$, there exists a subset $T^* \subseteq C$ such that (1) $|T^* \cap C_{i,j}| \geq \delta_{i,j}(f)$ for all $i, j \in [m]$ with $i \leq j$, (2) $|T^*| = \mathsf{opt}$, and (3) $\{c_{k_1}, \ldots, c_{k_t}\} \subseteq T^*$.*

Proof: We prove the observation by induction on t. For $t = 0$, the statement trivially holds. Suppose the statement holds for $t-1$, i.e., there exists a subset $T^* \subseteq C$ satisfying the first two conditions in Lemma 13 and $\{c_{k_1}, \ldots, c_{k_{t-1}}\} \subseteq T^*$. We show the statement holds for t. Specifically, we shall modify T^* to make it satisfy $\{c_{k_1}, \ldots, c_{k_t}\} \subseteq T^*$ while maintaining the first two conditions in the observation.

First, we notice that T^* must contain a point other than $c_{k_1}, \ldots, c_{k_{t-1}}$. To see this, suppose $T^* = \{c_{k_1}, \ldots, c_{k_{t-1}}\}$. Since our algorithm added c_{k_t} to T, there exist $i, j \in [m]$ with $i \leq k_t \leq j$ such that $\delta_{i,j}(f) \geq |\{c_{k_1}, \ldots, c_{k_{t-1}}\} \cap C_{i,j}| + (j - k_t + 1)$. This implies that $\delta_{i,j}(f) > |\{c_{k_1}, \ldots, c_{k_{t-1}}\} \cap C_{i,j}|$, that is, $\delta_{i,j}(f) > |T^* \cap C_{i,j}|$, which contradicts the fact that T^* satisfies the first condition in the lemma. Thus, T^* contains a point other than $c_{k_1}, \ldots, c_{k_{t-1}}$.

Now let k be the smallest index such that $c_k \in T^* \backslash \{c_{k_1}, \ldots, c_{k_{t-1}}\}$. If $k = k_t$, then $\{c_{k_1}, \ldots, c_{k_t}\} \subseteq T^*$ and we are done. Otherwise, we remove c_k from T^* and add c_{k_t} to T^*. After this modification, it is clear that $|T^*| = \mathsf{opt}$ and $\{c_{k_1}, \ldots, c_{k_t}\} \subseteq T^*$. So it suffices to show $|T^* \cap C_{i,j}| \geq \delta_{i,j}(f)$ for all $i, j \in [m]$ with $i \leq j$. Consider indices $i, j \in [m]$ with $i \leq j$. By assumption, before the modification we have $|T^* \cap C_{i,j}| \geq \delta_{i,j}(f)$. If $j \geq k_t$, then $|T^* \cap C_{i,j}|$ does not decrease after the modification, and is thus at least $\delta_{i,j}(f)$. So assume $j < k_t$. In this case, $T \cap C_{i,j} = \{c_{k_1}, \ldots, c_{k_{t-1}}\} \cap C_{i,j} \subseteq T^* \cap C_{i,j}$. Since $|T \cap C_{i,j}| \geq \delta_{i,j}(f)$ by Lemma 9, we have $|T^* \cap C_{i,j}| \geq \delta_{i,j}(f)$. ■

We use binary search to find the smallest r such that the reduced instance has an f-tolerant hitting set of size at most k. Therefore, the following theorem holds.

Theorem 13 *Optimal Fault-Tolerant Committee can be solved in time* $\mathcal{O}(n^2m^3f\log(nm))$ *for one-dimensional Euclidean elections.*

Remark 1 *Our dynamic programming algorithm works as long as either the set V or the set C is embedded in $\mathbb{R}$ (i.e., has a linear ordering), while the other set can have an arbitrary d-dimensional embedding. Moreover, we can also extend our algorithms to ordinal elections with (widely studied) single-peaked preferences [1, 42, 43, 44] to compute an optimal fault-tolerant Chamberlin-Courant committee.*

Chapter 4

FTCS in Multidimensional Instances

In the previous chapter, we introduced a new notion of fault-tolerance in committee selection and we studied the fault-tolerant committee selection problems for one-dimensional instances (i.e., when candidates and voters lie on a line). Although we presented polynomial-time algorithms for all our problems, the problem was already nontrivial in 1D. In this chapter, we will consider fault-tolerance in multidimensional instances where candidates and voters lie in a d-dimensional Euclidean space and voters' preferences are derived based on the Euclidean distances. In particular, we will investigate all three problems – Optimal Replacement Problem (ORP), Fault-tolerance Score (FTS) and Optimal Fault-Tolerant Committee (OFTC) in multidimensions. We refer the reader to Subsection 3.0.1 for the formal problem statements.

4.0.1 Results and Organization of the Chapter

Before discussing our results for multidimensional instance, we briefly recall our notations from the last chapter. A Euclidean election is represented by $E = (\mathcal{C}, V)$. For a committee $T \subseteq \mathcal{C}$, the score of a voter for a committee is defined as $\sigma(v, T) = \min_{c \in T} d(v, c)$, and the score of T as $\sigma(T) = \max_{v \in V} \sigma(v, T)$. f is the fault-tolerance (integer) parameter, a failing set is denoted by J and a replacement set is denoted by R. For an integer j, the j-fault-tolerance score of T is denoted by $\sigma_j(T) = \max_{J \subseteq C \text{ s.t. } |J| \leq j} \sigma((T \setminus J) \cup R)$. We now summarize our results.

In two dimensions and higher, OFTC is NP-hard because of its close connection to the k-center problem. However, we show that even the seemingly simpler problem of optimal replacement (ORP) is also NP-hard (see Section 4.1 for details). Our main results include a constant-factor approximation for all three problems in any fixed dimension (in fact, in any metric space) which we present in Section 4.2, as well as a novel bicriterion FPT approximation via an EPTAS whose running time has the form $f(\epsilon)n^{\mathcal{O}(1)}$ presented in Section 4.3. For a complete set of results and for ease of reference, we show all our results (including the ones presented in Chapter 3) on fault-tolerant committee selection problems in the following table.

	One-dimensional instances	**Dimension $d \geq 2$**		
		Complexity	**Approximation**	**Bounded f**
ORP	P (Theorem 11)	NP-hard (Theorem 14)	3-approx. (Lemma 19)	P (Section 4.1.3)
FTS	P (Theorem 12)	NP-hard (Theorem 14)	3-approx. (Lemma 20)	P (Section 4.1.3)
OFTC	P (Theorem 13)	NP-hard (Theorem 14)	5-approx. (Lemma 23) Bicriterion-EPTAS (Theorem 16)	NP-hard (Section 4.1.3) 3-approx. (Theorem 15)

Table 4.1: Summary of our results.

4.1 NP-hardness Results

In this section, we settle the complexity of all three problems in multidimensional elections. Unsurprisingly, the optimal committee design problem is intractable — it is similar to facility location — but it turns out that the seemingly simpler variants ORP

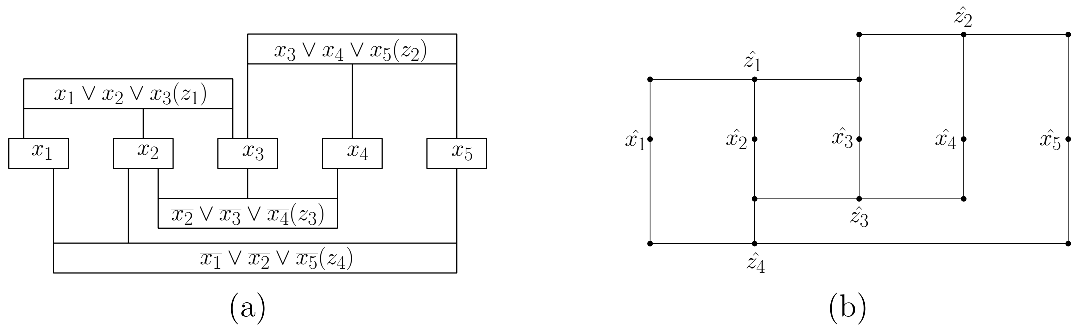


Figure 4.1: Figure (a) shows rectangular embedding of the PM-3SAT instance given as a part of the input. In figure (b), we show a transformation of rectangular embedding to orthogonal embedding useful for our construction. Here, each variable x_i in the rectangular embedding is replaced by the variable reference point $\hat{x}_i$ and each clause is replaced z_i is replaced by the clause reference point $\hat{z}_i$.

and FTS are also intractable.

Theorem 14 *All three problems (Optimal Replacement, Fault-Tolerance Score, and Optimal Fault-Tolerant Committee) are NP-hard, in any dimension $d \geq 2$ under the Euclidean norm, where size of the committee k and the failure parameter f are part of the input.*

We will use a single construction to show NP-hardness all three problems. Our proof uses a reduction from the NP-complete problem Planar Monotone 3-SAT (PM-3SAT) [16]. An input to this problem is a *monotone* 3-CNF formula ϕ where each clause contains either three positive literals or three negative literals, and whose variable-clause incidence graph has a planar embedding which is given as a part of the input. Given an instance ϕ of PM-3SAT, our reduction constructs a 2-dimensional Euclidean election. The general outline follows a scheme used in Section 2.1 to show the hardness of committee selection under *ordinal* preferences, but generalizing the proof to *fault-tolerant committees* requires several technical modifications and a new proof of correctness.

For ease of referencing, we adapt the terminologies from Section 2.1. In the planar embedding of the formula ϕ, each variable/clause is drawn as an (axis-parallel) rectangle in the plane, and so this is called a *rectangular embedding*. See Figure 4.1a for an illustration. The rectangles for the variables are drawn along the x-axis, while the rectangles for the positive (resp., negative) clauses lie above (resp., below) the x-axis. If a clause contains a variable, then there is a vertical segment connecting the clause rectangle and the variable rectangle. Each such vertical segment is disjoint from all the rectangles except the two it connects.

The rectangular embedding of ϕ can be modified to another embedding which is easier to work with called *orthogonal embedding*. We refer the reader to [20] for details of the modification (See figure 4.1b for intuition). The intersection points of vertical and horizontal segments in the orthogonal embedding are *connection points*. To build the intuition for the orthogonal embedding, we now give its properties as stated in Section 2.1:

(i) Vertical and horizontal segments do not cross.

(ii) Each horizontal segment corresponds to a clause in ϕ. Moreover, it intersects exactly three vertical segments corresponding to the literals in that clause.

(iii) The endpoints of all segments are connection points.

For each horizontal segment, we will refer to the middle connection point as the *reference point* of a the clause (notice that from properties (ii) and (iii) each horizontal segment has three connection points and two of those are the left and the right endpoint of the segment). The reference points for the variables $x_1, \ldots, x_n$ are denoted by $\hat{x_1}, \ldots, \hat{x_n}$ and for the clauses $z_1, \ldots, z_m$ they are denoted by $\hat{z_1}, \ldots, \hat{z_m}$. With shifting/scaling of the orthogonal embedding without changing its topology, we can ensure that the x-coordinates (y-coordinates) of vertical (resp., horizontal) segments are distinct even integers in the range $\{1, \ldots, 2n\}$ (resp., $\{-2m, \ldots, 2m\}$). This guarantees that all connection points have even integer coordinates and the embedding is contained in $[1,2n] \times [-2m,2m]$ rectangle. Now using the integral points on each segment s, we can partition s into $\ell(s)$ parts each of unit length where $\ell(s)$ is the length of s. These unit length segments are called *pieces* of the orthogonal embedding. We use N to denote the total number of pieces. Note that $N = \mathcal{O}(nm)$.

We now construct a Euclidean election $E = (V, C)$ with voters and candidates as points in $\mathbb{R}^2$.

Variable gadgets. For each variable x_i, we choose two additional points near (but not equal to) x_i as follows. Recall that there are two vertical pieces incident to $\hat{x}_i$ in the orthogonal embedding: one above $\hat{x}_i$, and the other below $\hat{x}_i$. We choose a point with distance 0.2 from $\hat{x}_i$, on each of the two pieces. Next, we put $f+1$ candidates on each of these two points and a (single) candidate at $\hat{x}_i$ (we set the value of f later in the construction). Furthermore, we place a voter on each of these three points. We call these candidates/voters the x_i-*gadget*. See figure 4.2a. For $i \in [n]$, we construct the x_i-gadget for each variable x_i. Overall, the variable gadgets have $(2f+3)n$ candidates and $3n$ voters.

Clause gadgets. Next, we construct a set of candidates/voters for the clauses $z_1, \ldots, z_m$. For each clause z_i, we put a voter at the reference point $\hat{z}_i$, and call this voter the z_i-*gadget*. See figure 4.2b. The total number of voters in the clause gadgets is m. Clause gadgets do not have any candidates.

Piece gadgets. Finally, we construct a set candidates/voters to connect the variable and clause gadgets. Consider a piece s of the orthogonal embedding. Recall that s is a unit-length segment. Let s^- and s^+ be the two endpoints of s. We identify these endpoints as follows: For a vertical piece s above (resp., below) the x-axis, we say s^- is the bottom (resp., top) endpoint of s and s^+ is the top (resp., bottom) endpoint of s. For a horizontal piece s, s must belong to the horizontal segment of some clause z_i. Suppose s is to the left (resp., right) of the reference point $\hat{z}_i$, then s^- is the left (resp., right) endpoint of s and s^+ be the right (resp., left) endpoint of s. For every piece s that is *not* adjacent to any clause reference point, we choose four points near s^+ and add candidates/voters on them as follows: We place $f+1$ candidates each on a point which is 0.3 below and 0.3 to the right of s^+, and on a point which is 0.4 above and 0.3 to the left of s^+, and on a point at s^+. Further, we place a candidate at a point which is 0.3 above s^+. Lastly, we place one voter at each of these four points. We call these the candidates/voters of the *s-gadget*. See figure 4.2c. Note that pieces adjacent some clause reference points do not have gadgets. Therefore, the total number of candidates in the piece gadgets is $(3f+4)(N-3m)$, as each clause reference point is adjacent to three pieces, and the number of voters is $4(N-3m)$.

Combining these three constructed gadgets, our election $E=(C,V)$ has $4N-11m+3n$ voters and $(3N-9m+2n)f+4N-12m+3n$ candidates. We set the committee size k equals to $N+n-3m$. Clearly, the construction can be done in a polynomial time. The main intuition behind the construction is the following.

Candidates in the constructed instance can be partitioned into two types:

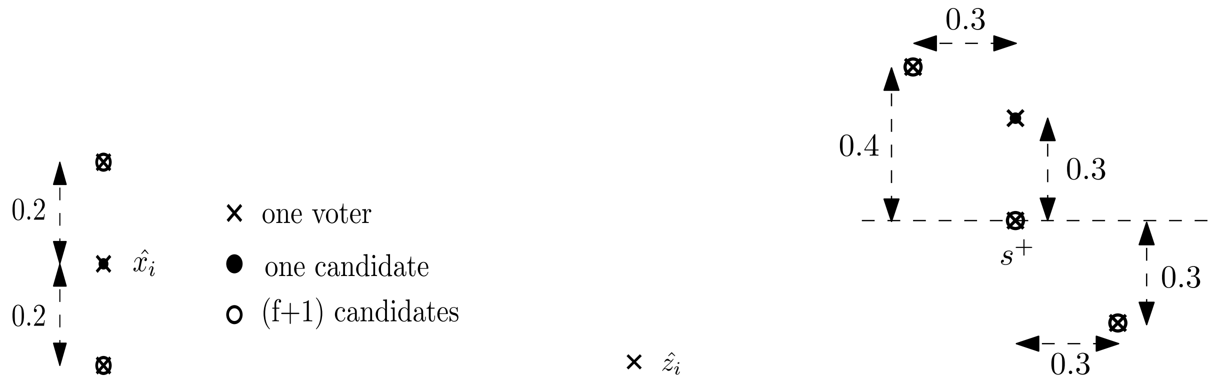
- *Robust candidates* ($C_{rob} \subseteq C$) is the set of candidates such that for each candidate, there are f other candidates at the exact same location as it. Note that for any failing set $J \subseteq C$, at least one candidate is live in each of these locations.

- *Covering candidates* ($C_{cov} \subseteq C$) is the set of candidates such that for each candidate in the set, it is the unique candidate at its location. Note that $|C_{cov}| = k$.

In the constructed election, the following lemma holds.

Lemma 14 *In the constructed election $E=(C,V)$, we have $\sigma(C_{cov}) \leq 0.75$ where $C_{cov} \subseteq C$ is the set of covering candidates.*

Proof: For all voters $v \in V$, we will show that $d(v, C_{cov}) \leq 0.75$.



(a) Vertex gadget (b) Edge gadget (c) Piece gadget

Figure 4.2: Gadgets in our construction. Here, a disk and a cross at the same location indicates a voter and a candidate at the same location. Similarly, a circle and a cross at the same location indicates a voter and $(f+1)$ candidates at the same location.

First, we consider the voters in the variable gadget. For a variable x_i, the candidate at $\hat{x}_i$ belongs to C_{cov}. All three voters in the variable gadget are within distance 0.2 from $\hat{x}_i$. Therefore, all the voters in the variable gadget have $d(v, C_{cov}) \leq 0.75$.

Next, we consider the clause-gadgets. The closest candidate in T for a voter placed at $\hat{z}_i$ is at a distance 0.7 from it; therefore, all voters in the clause gadgets all have $d(v, C_{cov}) \leq 0.75$.

Finally, we consider the piece gadgets. For each piece s, C_{cov} contains a candidate at a distance 0.3 above s^+. It can be verified that all four voters in the piece gadget have their closest candidate in C_{cov} within distance $\sqrt{0.45} < 0.7$ (See figure 4.2c).

Hence, for all voters in the piece gadgets, $d(v, C_{cov}) \leq 0.75$. This completes the proof of Lemma 14. ■

The election constructed above will be used to show hardness for all three problems. Let the distance $r = 0.75$. Recall that $k = N + n - 3m$. We will now, we describe the decision version of each of our problems along with the construction of additional elements necessary in their input:

(i) **ORP**: In the input of ORP we additionally need a committee $T \subseteq C$ and a failing set $J \subseteq C$. We set $T = J = C_{cov}$.

We ask if there exists a replacement $R \subseteq C \setminus C_{cov}$ such that $\sigma(R) \leq r$?

(ii) **FTS**: In the input of FTS we additionally need a committee $T \subseteq C$ and a fault-tolerance parameter f. We set $T = C_{cov}$ and $f = k$.

We ask if $\sigma_f(T) \leq r$?

(iii) **OFTC**: In the input of OFTC we additionally need a committee size k' and a fault-tolerance parameter f. We set $k' = k$ and $f = k$.

We ask if there exists a k-sized committee $T \subseteq C$ with $\sigma_f(T) \leq r$?

To show the equivalence, we will show that the answer to each of the above question is yes if and only if ϕ is satisfiable.

The following lemma shows the proof of equivalence for problem (i).

Lemma 15 *There exists a committee $R \subseteq C \setminus C_{cov}$ with size k such that $\sigma(R) \leq r$ if and only if ϕ is satisfiable where $k = |C_{cov}| = N + n - 3m$, N is the total number of pieces, and m (resp., n) is the number of clauses (resp., variables) in ϕ, respectively.*

In the next two subsections, we give the proof of Lemma 15.

4.1.1 Forward direction of Lemma 15

In this section, we will show that if ϕ is satisfiable, then there exists a k-sized committee $R \subseteq C \setminus C_{cov}$ such that $\sigma(R) \leq r$ (recall that $r = 0.75$).

Suppose ϕ is satisfiable. Let $\pi : X \rightarrow \{\mathsf{true}, \mathsf{false}\}$ be an assignment which makes ϕ true. We construct a k-sized committee $R \subseteq C \setminus C_{cov}$ with $\sigma(R) \leq r$ using ϕ. We include one candidate from the every variable gadget and every piece gadget to R as follows:

- *Replacement candidates from variable gadgets:* Consider a variable x_i. By our construction, the x_i-gadget contains $2f + 3$ candidates which have the same x-coordinates as $\hat{x}_i$. If $\pi(x_i) = \mathsf{true}$ (resp., $\pi(x_i) = \mathsf{false}$), we include in R one of the topmost (resp., bottommost) candidates in the x_i-gadget.

- *Replacement candidates from piece gadgets:* Consider a piece s not adjacent to a clause reference point (recall that pieces adjacent to some clause reference point do not have gadgets on them). We begin by defining a variable as an *associated* variable of s in the same way as described in Section 2.1: When s is vertical, the associated variable of s is just the variable whose vertical segment contains s. For when s is horizontal, then s must belong to the horizontal segment of some clause z_j. Then, if s to the left of the reference point $\hat{z}_j$ then the associated variable of s is the variable whose vertical segment intersects the left endpoint of the horizontal segment of z_j, and vice versa when s to the right of the reference point $\hat{z}_j$.

  Let x_i be the associated variable of s. Then,

  (i) If $\pi(x_i) = \mathsf{true}$: We include in R a candidate in the s-gadget that is 0.4 above and 0.3 to the left (resp., 0.4 below and 0.3 to the right) of s^+ if s is above (resp., below) the x-axis.

  (ii) If $\pi(x_i) = \mathsf{false}$: We include in R a candidate in the s-gadget that is 0.3 below and 0.3 to the right (resp., 0.3 above and 0.3 to the left) of s^+ if s is below (resp., above) the x-axis.

This finishes the construction of the committee R. Recall that the total number of variable and the piece gadgets is $N + n - 3m = k$. Therefore, $|R| = k$. The following lemma completes the "if" part of Lemma 15.

Lemma 16 *$\sigma(R) \leq r$.*

Proof: For all voters $v \in V$ in the constructed instance, we will show that $d(v, R) \leq r$.

First, we consider the voters in the variable gadget. For a variable x_i, either a candidate 0.2 above or 0.2 below $\hat{x}_i$ belongs to R. Hence, for all three voters, $d(v, R) \leq 0.6$. Recall that $r = 0.75$. Therefore, all the voters in the variable gadget have $d(v, T) \leq r$.

Next, we consider the clause-gadgets. For a clause z_i, consider the voter v placed at $\hat{z}_i$. The closest candidate in R for this voter, is the candidate placed at 0.4 above and 0.3 to the left of s^+ from a clause gadget with $d(\hat{z}_i, s^+) = 1$. Hence, $d(v, R) = \sqrt{0.45} < r$.

Finally, we consider the piece gadgets. We consider two cases: piece gadget not adjacent to a variable reference point and piece gadget adjacent to a variable reference point. Let x_i be the associated variable with the piece gadget s, and $\pi(x_i) = \mathsf{true}$ (the case when $\pi(x_i) = \mathsf{false}$ is symmetric). Let s be a piece gadget not adjacent to any variable reference point. Suppose s is above x-axis, and $\hat{s}$ is the gadget below s (the case when s is below the x-axis is symmetric; hence, we leave the proof for that to the reader). Recall that the piece gadgets contains four voters: Let v_1, v_2, v_3, and v_4 be the voters placed at distances 0.4 above and 0.3 to the left of s^+, 0.3 above s^+, at s^+, and 0.3 below and 0.3 to the right of s^+, respectively. We know that a candidate placed at the location of voter v_1 (say c_1) belongs to R. Notice v_1, v_2 and v_3 have a distance at most 0.5 from this voter but v_4 has a distance $\sqrt{0.85} > r$. But consider the gadget corresponding to $\hat{s}$. Here, we know R contains a candidate (say c'_1) placed at 0.4 above and 0.3 to the left of $\hat{s}^+$. It can be verified that $d(v_4, c'_1) = \sqrt{0.45} < r$. Therefore, $d(v_4, R) \leq r$. We now consider the case when s is adjacent to a variable reference points (say $\hat{x}_i$). We know s has four voters v_1, v_2, v_3, and v_4 placed at distances 0.4 above and 0.3 to the left of s^+, 0.3 above s^+, at s^+, and 0.3 below and 0.3 to the right of s^+, respectively. We know that a candidate placed at the location of voter v_1 (say c_1) belongs to R, and this candidate is at a distance at most 0.5 from v_1, v_2, and v_3. Recall that $\pi(x_i) = \mathsf{true}$, R contains a candidate at a distance 0.2 above $\hat{x}_i$. Notice that this candidate is at a distance $\sqrt{0.34}$ from v_4. Therefore, $d(v_4, R) < r$. This completes the proof of Lemma 16. ■

4.1.2 Reverse direction of Lemma 15

Suppose $R \subseteq C \setminus C_{cov}$ is a k-sized committee with $\sigma(R) \leq r$. We will show how to recover a satisfying assignment $\pi : X \to \{\mathsf{true}, \mathsf{false}\}$ using R. The structure of our proof is similar to the reverse direction argument in Section 2.1. First, we observe the following property of R.

Lemma 17 *R contains exactly one candidate from every variable and piece gadget.*

Proof: We begin with the variable gadgets. Consider an x_i-gadget corresponding to the variable x_i. Recall that we place three voters in the x_i-gadget: One voter at $\hat{x}_i$ (say v_1), and one voter each at a distance 0.2 above and 0.2 below of $\hat{x}_i$. Observe that for v_1, its distance to any candidate from the adjacent piece gadgets is at least $\sqrt{(0.7)^2 + (0.3)^2} = \sqrt{0.58}$ which is strictly greater than r. Hence, R must include at least one candidate from the x_i-gadget to ensure $d(v_1, R) \leq r$. We now consider the piece gadgets. Let v_1 be the voter placed at s^+. The nearest candidate to v_1 from the adjacent piece gadgets is at a distance at least $\sqrt{0.58}$ which is strictly greater than r. Therefore, R must contain at least one candidate from each piece gadget to ensure $d(v_1, R) \leq r$.

Finally, recall that the committee size is $k = N + n - 3m$ and the number of variable (piece) gadgets is $n(N - 3m)$, respectively. Therefore, by a simple counting argument, we can conclude that T must contain exactly one candidate from each variable gadget and each piece gadget. This completes the proof of Lemma 17. ■

We will now use R to recover a satisfying assignment π for ϕ. For an arbitrary variable x_i, using Lemma 17, we know R contains exactly one candidate (say c_i) from the x_i-gadget. We set $\pi(x_i)$ as follows:

- If c_i is above the x-axis, we set $\pi(x_i) = \mathsf{true}$.
- If c_i is below the x-axis, we set $\pi(x_i) = \mathsf{false}$.

To complete the proof, we need to show that π is a satisfying assignment of ϕ. It is enough to show that for each clause, at least one of its variables is set to true. Since the argument for positive and negative clauses is similar, we will only that for each positive clause, at least one its variables is set to true. We begin by proving the following important structural property of the committee R.

Lemma 18 *For a piece s above the x-axis that is not adjacent to any clause reference point, suppose x_i is the associated variable of s. If R contains the candidate in the s-gadget which is 0.4 above and 0.3 to the left of s^+, then $\pi(x_i) = \mathsf{true}$.*

Proof: Let s be piece as described in the above lemma with the associated variable x_i. We will show that if R contains the candidate in the s-gadget which is 0.4 above and 0.3 to the left of s^+, then R contains a candidate 0.2 above $\hat{x}_i$ in the x_i-gadget. The choice of candidate in R from the s-gadget to the x_i-gadget percolates as follows:

- When s is not adjacent to a variable reference point $\hat{x}_i$: Let v_1, v_2, v_3, and v_4 be the voters placed at distances 0.4 above and 0.3 left of s^+, 0.3 above s^+, at s^+, and 0.3 below and 0.3 to the right of s^+, respectively. We know that the candidate placed at s^+ belongs to J. From the set of $f+1$ candidates placed at the locations of v_1, v_2 and v_4, we denote a candidate by c_1, c_2, and c_3, respectively. Moreover, let s' be the piece below (resp., to the left of) s when s is a vertical (resp., horizontal) piece. We denote the corresponding candidate in s' by c'_1, c'_2, and c'_3. Assume that $c_1 \in R$. We observe that $d(v_4, c_1) = \sqrt{0.85} > r$ but $d(v_4, c'_1) = \sqrt{0.45} < r$. Moreover, c'_1 is the only alive candidate from s'-gadget within distance r from v_4. Using Lemma 2, we know R only includes c_1 from s. Hence, to satisfy $d(v_4, R) \leq r$, R must include the candidate c_1'. Observe that we can repeat the above argument for all pieces below (resp., to the left of) s, which implies that for all pieces s_i below (resp., to the left of) s, R includes the candidate in s_i-gadget placed 0.4 above and 0.3 to the left of s_i^+.

- When s is adjacent to a variable reference point $\hat{x}_i$: Let v_1, v_2, v_3, and v_4 be the voters placed at distances 0.4 above and 0.3 left of s^+, 0.3 above s^+, at s^+, and 0.3 below and 0.3 to the right of s^+, respectively. Moreover, let c_1, c_2, and c_3 denote an arbitrary candidate placed at the locations of voters v_1, v_2, and v_4, respectively. We know that for voter v_4, the set of candidates within distance r is $\{c_2, c_3, c_4\}$ where c_4 is the candidate placed 0.2 above $\hat{x}_i$. Using Lemma 17, we know R only includes c_1 from the piece $\hat{s}$. Hence, to ensure $d(v_4, R) \leq r$, R must include candidate c_4. Recall that for a variable x_j for $j \in [n]$, if R includes a candidate above the reference point $\hat{x}_j$, we set $x_j = \textsf{true}$. Since, $c_4 \in R$ and c_4 lies above $\hat{x}_i$, we set $x_i = \textsf{true}$.

This completes the proof of Lemma 18. ■

Using Lemma 17 and Lemma 18 we are now ready to show that the constructed assignment π satisfies ϕ. Since the argument for positive and negative clauses is similar, we will only show for the positive clauses. Our argument is similar to the one in Section 2.1 but we include it here for completeness.

Consider a positive clause z_i. We will show that at least one variable of z_i is set to $\textsf{true}$ by π. We denote the pieces adjacent to the reference point $\hat{z}_i$ by s_1, s_2, s_3. Without loss of generality, let s_1 be to the left of $\hat{z}_i$, s_2 be to the right of $\hat{z}_i$, and s_3 be below $\hat{z}_i$. Notice that $\hat{z}_i = s_1^+ = s_2^+ = s_3^+$. Recall that $\hat{z}_i$ is a connection point. Since all connection points have even coordinates and s_1^-, s_2^-, s_3^- are at a unit distance from $\hat{z}_i$, s_1^-, s_2^-, s_3^- are not connection points. Hence, let s_4, s_5, s_6 be the pieces such that the right endpoint of s_4 is s_1^-, the left endpoint of s_5 is s_2^-, and the top endpoint of s_6 is s_3^-. Therefore, $s_1^- = s_4^+$,

$s_2^- = s_5^+$, and $s_3^- = s_6^+$. Let c_4 be a candidate in s_4-gadget such that c_4 is 0.4 above and 0.3 to the left of s_4^+. Moreover, let the candidates c_5 and c_6 defined in the similar way such that c_5 belongs to the s_5-gadget and c_6 belongs to the s_6-gadget. For the voter at the reference point $\hat{z}_i$, only the candidates c_4, c_5, c_6 are within distance r. This is because all pieces except $s_1, \ldots, s_6$ have distances at least 2 from $\hat{z}_i$. Since $\sigma(R) \leq r$, R must contain at least one of these three candidates. Therefore, using Lemma 18, we conclude that at least one of the associated variables of s_4, s_5, s_6 is true. Since these are exactly the three variables in clause z_i; hence, z_i is true under the assignment π. This completes the "only if" part of our proof.

The argument above completes the proof of Lemma 15 and completes the argument of equivalence for our decision problem (i).

Argument of equivalence for the decision problem (ii)**:** Recall that for the FTS decision problem instance stated above, the input committee is $T = C_{cov}$ and the fault-tolerance parameter is $f = k$. Notice that T contains one candidate from each vertex and each piece gadget. Suppose a subset $J \subset C$ with $|J| \leq f$ fails. Consider the committee $T' = T \backslash J$. All voters in the vertex gadgets and the piece gadgets which have a non-empty intersection with T' have a committee member within distance r (i.e., suppose $V' \subseteq V$ is the subset of all such voters then $\sigma(V', T') \leq r$). For the voters in $V \setminus V'$, we build a committee R using the candidates in vertex and piece gadgets with have an empty intersection with T' in the same way as we constructed a replacement committee in the forward direction of proof of Lemma 15 (Section 4.1.1) (to replace the candidates from set $T \cap J$). Notice that it is always possible to construct such a replacement R because all candidates in R are robust candidates (meaning that there are $f + 1$ identical copies of each of these candidates) and the failure set J has size at most f.

Since the total number of vertex and edge gadgets is k, the new committee $(T \setminus J) \cup R$

has size k. Using the same argument as in Section 4.1.1, we can show that $\sigma((T \setminus J) \cup R) \leq r$. This completes the argument in the forward direction (that is, if ϕ is satisfiable then $\sigma_f(T) \leq r$).

To show the reverse direction, suppose $\sigma_f(T) \leq r$. Therefore, when failing set $J = C_{cov}$, there exists a k-sized replacement $R \subseteq C_{rob}$ such that $\sigma(R) \leq r$. Hence, using Lemma 15, we can conclude that ϕ is satisfiable.

Argument of equivalence for the decision problem (iii)**:** Recall that for the OFTC decision problem instance stated above, the input is the committee size $k = N + n - 3m$ and the fault-tolerance parameter $f = k$.

The argument for the forward direction is trivial, as we know when ϕ is satisfiable, the k-sized committee $T = C_{cov}$ has $\sigma_f(T) \leq r$. In the reverse direction, suppose $T \subseteq C$ is a k-sized committee with $\sigma_f(T) \leq r$. Therefore, in this case, for a size f failing set $J = C_{cov}$, there exists a replacement $R \subseteq C \setminus C_{cov}$ such that $|(T \setminus J) \cup R| \leq k$ and $\sigma((T \setminus J) \cup R) \leq r$. Since $(T \setminus J) \cup R = R$ and $R \subseteq C_{rob}$; using Lemma 15, we can conclude that ϕ is satisfiable.

4.1.3 Hardness when f is bounded

ORP and FTS Consider an election $E = (C, V)$, a committee $T \subseteq C$ of size k and a fault-tolerance parameter f which is a constant. It is easy to see that we can solve ORP optimally in time $nm^{\mathcal{O}(f)}k$ by trying all possible replacement sets and choosing the best one. Similarly, by trying all possible failing sets of size at most f (note that there are $m^{\mathcal{O}(f)}$ such sets) and computing optimal replacement for each set, we can compute $\sigma_f(T)$ in time $nm^{O(f)}k$.

OFTC We will now show that for any integer $f \geq 0$, OFTC is NP-hard using a simple reduction from the k-supplier problem [45]: Fix $f \geq 0$. Let $(\mathcal{C}, F)$ along with an integer k be a k-supplier instance where $\mathcal{C}$ is the set of customers and F is a set of facilities embedded in $\mathbb{R}^2$. In the decision version of k-supplier, given a real number r, we ask if there exists a size k set $F' \subseteq F$ such that $\sigma(\mathcal{C}, F') \leq r$.

We construct an election $E = (C, V)$ in $\mathbb{R}^2$ as follows. We set $V = \mathcal{C}$. Further, we construct the set of candidates C by adding $f + 1$ identical candidates on each point in F. We set the committee size to k. It is easy to see that there exists a k-sized committee $T \subset C$ with $\sigma_f(T) \leq r$ if and only if there exists a k-sized subset $F' \subseteq F$ such that $\sigma(\mathcal{C}, F') \leq r$ where r is a real number.

4.2 Constant factor Approximations

In the previous section, we showed that all three problems are NP-hard. We now turn to approximation algorithms. In particular, we give 3-approximation algorithms for ORP and FTS in subsections 4.2.1 and 4.2.2, respectively. Next, using a nontrivial packing argument, we give a 3-approximation algorithm for OFTC in Subsection 4.2.3 which is followed by a simple 5-approximation algorithm. All of these approximations hold not just for d-dimensional Euclidean space, for any fixed d, but also for any metric space.

4.2.1 Optimal Replacement Problem

A simple greedy algorithm achieves a 3-approximation for the Optimal Replacement Problem in any fixed dimension d as well as in any metric space.

Lemma 19 *We can find a 3-approximation for ORP in time $O(k(nk + m))$.*

Proof: Let $T \subseteq C$ be the given committee and let $J \subseteq T$ be the failing set.

In order to find the replacement set R, we initialize $\hat{T} = T \setminus J$, and then repeat the following two steps $|T \cap J|$ times: (1) Choose the farthest voter from $\hat{T}$, namely, choose $\hat{v} = \arg\max_{v \in V} d(v, \hat{T})$, and (2) Add to $\hat{T}$ the candidate $\hat{c} \notin \hat{T}$ that is closest to $\hat{v}$. Upon termination, we clearly have $|\hat{T}| = |T|$.

We will now show that R is a 3-approximate replacement committee. Let $|T \cap J| = r$. Suppose $R^* = \{c_1^*, c_2^*, \ldots, c_r^*\}$ is an optimal replacement such that $T^* = (T \setminus J) \cup R^*$ has $\sigma(T^*) = \sigma^*$. Let $V_1^*, V_2^*, \ldots, V_r^* \subseteq V$ be disjoint set of voters such that c_i^* is the closest candidate in T^* for all voters in V_i^* for $i \in [r]$. We define $V' = \bigcup_{i=1}^{r} V_i^*$.

Let $R = \{c_1, c_2, \ldots, c_r\}$ be the replacement set constructed by our algorithm and let $\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_r$ be the voters chosen by our algorithm. Recall that $\hat{T} = (T \setminus J) \cup R$. It is easy to see that $\sigma(V \setminus V', T^*) = \sigma(V \setminus V', \hat{T}) \leq \sigma^*$ since $(T \setminus J) \subseteq T^*$ and $(T \setminus J) \subseteq \hat{T}$. Next, using a simple case analysis we will show that $\sigma(V', \hat{T}) \leq 3\sigma^*$.

Our cases are based on the voters $\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_r$ as follows:

- If $\hat{v}_i \in V \setminus V'$ for $i \in [r]$, then the farthest voter (aka $\hat{v}_i$) in the i^{th}-iteration of the algorithm satisfies $\sigma(\hat{v}_i, \hat{T}) \leq \sigma(V \setminus V', \hat{T}) \leq \sigma^*$, and hence, $\sigma(V', \hat{T}) \leq \sigma^*$.

- Next, suppose for some $i, j, k \in [r]$, we have $\hat{v}_i, \hat{v}_j \in V_k^*$. Without loss of generality, let $j > i$. Then, in the j^{th}-iteration, the farthest voter (aka $\hat{v}_j$) among all voters in V has $d(\hat{v}_j, \hat{T_{j-1}}) \leq 3\sigma^*$ where $\hat{T_{j-1}}$ is a committee after $j-1$ iterations of adding replacement candidates. This is because $d(\hat{v}_j, \hat{T_{j-1}}) \leq d(\hat{v}_j, \hat{v}_i) + d(\hat{v}_i, \hat{c}_i)$, and we know $d(\hat{v}_j, \hat{v}_i) = d(\hat{v}_j, c_i^*) + d(c_i^*, v_i) \leq 2\sigma^*$ and $d(\hat{v}_i, \hat{c}_i) \leq \sigma^*$ since c_i is the closest candidate to v_i which is not in the committee. This implies that $\sigma(\hat{T_{j-1}}) \leq 3\sigma^*$. Since $\hat{T_{j-1}} \subseteq \hat{T}$; therefore, $\sigma(V', \hat{T}) \leq 3\sigma^*$.

- Finally, we consider the case for when $i \in [r]$, we have $\hat{v}_i \in V_i^*$. We will now show that for $v \in V_i^*$, $\sigma(v, \hat{T}) \leq 3\sigma^*$. We know $d(v, \hat{c}_i) \leq d(v, \hat{v}_i) + d(\hat{v}_i, \hat{c}_i)$. Notice that

$d(v, \hat{v}_i) \leq d(v, c_i^*) + d(c_i^*, \hat{v}_i) \leq 2\sigma^*$, and $d(\hat{v}_i, \hat{c}_i) \leq \sigma^*$. Therefore, $d(v, \hat{c}_i) \leq 3\sigma^*$. This implies, $\sigma(V', R) \leq 3\sigma^*$, and hence, $\sigma(V', \hat{T}) \leq 3\sigma^*$.

Finally, it is easy to see that the algorithm runs for at most k iterations and each iteration can be trivially implemented in time $O(nk + m)$. This completes the proof of Lemma 19.

■

4.2.2 Computing the Fault-Tolerance Score

We can also approximate the optimal fault-tolerance score of a committee within a factor of 3. Specifically, if the optimal fault-tolerance score of T is $\sigma_f(T) = \sigma^*$, then our algorithm returns a real number σ' such that $\sigma^* \leq \sigma' \leq 3\sigma^*$.

For each voter v, let $d_f(v)$ be v's distance to its $(f+1)^{th}$ closest candidate, and let $d' = \max_{v \in V} d_f(v)$ be the maximum of these values over all voters. The basic idea behind our approximation is simple and uses the following two facts: (1) $\sigma^* \geq d'$, and (2) $\sigma^* \geq \sigma(T)$. The first one holds because d' is the best score possible if some voter's f nearest candidates fail, and the second one holds because a failure can only worsen the score (that is, $\sigma_f(T) \geq \sigma(T)$ for any $f > 0$). Therefore, the distance $\sigma' = d' + 2\sigma(T)$ is clearly within a factor of 3 of the optimal σ^*. We claim that for any failing set $J \subseteq C$, there exists a replacement $R \subseteq C \setminus J$ of size at most $|T \cap J|$ such that $\sigma((T \setminus J) \cup R) \leq \sigma'$.

Claim 2 *For a committee $T \subseteq C$ and a failing set $J \subseteq C$, there exists a replacement $R \subseteq C \setminus J$ of size at most $|T \cap J|$ such that $\sigma((T \setminus J) \cup R) \leq \sigma'$ where $\sigma' = d' + 2\sigma(T)$.*

Proof: Let $T \cap J = \{c_1, \dots, c_r\}$ and let $V_1, \dots, V_r$ be disjoint sets of voters such that c_i is the closest candidate in T to all voters in V_i for $i \in [r]$. We define $\overline{V} = \bigcup_{i=1}^{r} V_i$ and $V' = V \setminus \overline{V}$. For each voter $v \in V'$, its closest candidate in T is still available; hence, $\sigma(v, T \setminus J) \leq \sigma(T)$. Therefore, we only need to show that $\sigma(\overline{V}, R) \leq \sigma'$.

We build a replacement R as follows: We initialize $R = \emptyset$. Next, for r iterations, let i be the iteration index then,

- Select an arbitrary voter $v_i \in V_i$.
- Let $\hat{c}_i \in C \setminus J$ be the closest available candidate to v_i. Add $\hat{c}_i$ to R.

To show $\sigma(\overline{V}, R) \leq \sigma'$, we will now show that for all $i \in [r]$, $\sigma(V_i, \hat{c}_i) \leq \sigma'$. For $v \in V_i$, we know $d(v, \hat{c}_i) \leq d(v, v_i) + d(v_i, \hat{c}_i)$ where v_i is the voter selected in the i^{th}-iteration. First, observe that $d(v_i, \hat{c}_i) \leq d'$ as at most f closest candidates to v_i belong to J. Second, $d(v, v_i) \leq d(v, c_i) + d(c_i, v_i) \leq 2\sigma(T)$ (recall that $c_i \in T \cap J$). Hence, $d(v, \hat{c}_i) \leq 2\sigma(T) + d' = \sigma'$. Therefore, for the constructed replacement R, $\sigma(\overline{V}, R) \leq \sigma'$.

■

We have the following result.

Lemma 20 *The fault-tolerance score of a committee can be approximated within a factor of* 3 *in time* $\mathcal{O}(nm \log(f))$.

4.2.3 Optimal Fault-Tolerant Committee

We now discuss how to design approximately optimal fault-tolerant committees in multiwinner elections. Specifically, given a set of voters V and a set of candidates C in d-space, along with parameters k (committee size) and f (number of faults), we want to compute a size k committee $T \subseteq C$ with the minimum fault-tolerance score $\sigma_f(T)$. In this section, we prove the following result: We can solve this problem in polynomial time within an approximation factor of 3 in polynomial time if the parameter f is treated as a constant (while k remains possibly unbounded). If f is not assumed to be a constant, we can solve the problem within an approximation factor of 5.

Let σ^* be the optimal f-tolerant score of a committee of size k. We compute the approximation solution via an approximate decision algorithm, which takes as input a number $\sigma \geq \sigma_f(C)$ and returns a committee $T \subseteq C$ of size at most k with $\sigma_f(T) \leq 3\sigma$ if $\sigma \geq \sigma^*$. (We slightly abuse notation to introduce a convenient quantity $\sigma_f(C)$, which is the f-fault-tolerance score of a committee with all the input candidates. This is clearly a lower bound on any size k committee's score.)

For a committee $T \subseteq C$ and a failing set $J \subseteq C$, let $\delta(T, J)$ denote the score obtained after finding an optimal replacement K. That is,

$$\delta(T, J) = \min_{K \in C \setminus J, |K| = |T \cap J|} \sigma_0((T \setminus J) \cup K).$$

Thus, $\sigma_f(T) = \max_{J \subseteq C, |J| \leq f} \delta(T, J)$. Our approximation algorithm is shown in Algorithm 1. It begins with an empty committee T (line 1), and as long as there exists a failing set J of size at most f for which $\delta(T, J) > 3\sigma$, [1] we do the following.

First, we remove all candidates in J from T (line 3). Then, whenever there exists a voter $v \in V$ with $d(v, T) > 3\sigma$, we add to T a candidate $c \in C \setminus J$ whose distance to v is at most σ (lines 5-6). Such a c always exists because σ is at least the distance to the $(f+1)^{th}$ closest neighbor to v.

We call this voter v the *witness* of c, denoted by $\mathsf{wit}[c]$ (line 7). Adding c to T guarantees that $d(v, T) \leq \sigma$. We repeat this procedure (the inner while loop) until $d(v, T) \leq 3\sigma$ for all $v \in V$. Finally, the outer while loop terminates when $\delta(T, J) \leq 3\sigma$ for all $J \subseteq C$ of size at most f, i.e., $\sigma_f(T) \leq 3\sigma$. At this point, we return the committee T.

Lemma 21 *Let T be the committee computed by Algorithm 1. Then* $d(\mathsf{wit}[c], \mathsf{wit}[c']) > 2\sigma$

[1] We can check this condition by iterating over all failing sets of size f and computing an optimal replacement set in each case.

Algorithm 1 Approximate decision algorithm

Input: a set V of voters, a set C of candidates, the committee size k, the fault-tolerance parameter f, and a number $\sigma \geq \sigma_f(C)$

1: $T \leftarrow \emptyset$
2: **while** $\exists\ J \subseteq C$ such that $|J| \leq f$ and $\delta(T, J) > 3\sigma$ **do**
3:     $T \leftarrow T \backslash J$
4:     **while** $\exists\ v \in V$ such that $d(v, T) > 3\sigma$ **do**
5:         $c \leftarrow$ a candidate in $C \backslash J$ satisfying $d(v, c) \leq \sigma$
6:         $T \leftarrow T \cup \{c\}$
7:         $\mathsf{wit}[c] \leftarrow v$
8: **return** T

for any two distinct $c, c' \in T$.

Proof: Let $c, c' \in T$ such that $c \neq c'$. When the committee T is constructed in Algorithm 1, the candidates are added to T one by one (line 6). Therefore, without loss of generality, we can assume that c' is added to T after c. Consider the iteration of the inner while-loop (line 4-7) of Algorithm 1 in which we add c' to T. At the beginning of this iteration, we have $d(v, T) > 3\sigma$ where $v = \mathsf{wit}[c']$. Note that $c \in T$ at this time, and thus $d(\mathsf{wit}[c'], c) > 3\sigma$. Furthermore, we have $d(\mathsf{wit}[c], c) \leq \sigma$ by construction. Therefore,

$$d(\mathsf{wit}[c], \mathsf{wit}[c']) \geq d(\mathsf{wit}[c'], c) - d(\mathsf{wit}[c], c) > 2\sigma,$$

by the triangle inequality. ■

Lemma 22 *If $\sigma \geq \sigma^*$, then Algorithm 1 outputs a size k committee T with $\sigma_f(T) \leq 3\sigma$.*

Proof: The condition of the outer while loop of Algorithm 1 guarantees that $\delta(T, J) \leq 3\sigma$ for all $J \subseteq C$ of size at most f, which implies $\sigma_f(T) \leq 3\sigma$. To prove $|T| \leq k$, suppose $T = \{c_1, \ldots, c_r\}$. By Lemma 21, the pairwise distances between the voters $\mathsf{wit}[c_1], \ldots, \mathsf{wit}[c_r]$ are all larger than 2σ and thus larger than $2\sigma^*$ (as $\sigma \geq \sigma^*$ by our assumption). Now consider a committee $T^* \subseteq C$ of size k satisfying $\sigma_f(T^*) = \sigma^*$.

For each $\mathsf{wit}[c_i]$, there exists $c_i^* \in T^*$ such that $d(\mathsf{wit}[c_i], c_i^*) \leq \sigma^*$. Observe that $c_1^*, \ldots, c_r^*$ are all distinct. Indeed, if $c_i^* = c_j^*$ and $i \neq j$, then by the triangle inequality,

$$d(\mathsf{wit}[c_i], \mathsf{wit}[c_j]) \leq d(\mathsf{wit}[c_i], c_i^*) + d(\mathsf{wit}[c_j], c_j^*) \leq 2\sigma^*,$$

contradicting the fact that $d(\mathsf{wit}[c_i], \mathsf{wit}[c_j]) > 2\sigma^*$. Since $|T^*| = k$ and $c_1^*, \ldots, c_r^* \in T^*$, we have $r \leq k$, which completes the proof. ■

Using these two lemmas, we can compute a 3-approximate solution using Algorithm 1 as follows. First, we compute $\sigma_f(C)$ in $O(nm^{f+1})$ time by enumerating all failing sets $J \subseteq C$ of size at most f. For every voter $v \in V$ and every candidate $c \in C$ such that $d(v, c) \geq \sigma_f(C)$, we run Algorithm 1 with $\sigma = d(v, c)$. Among all the committees returned of size at most k, we pick the one, say T^*, that minimizes $\sigma_f(T^*)$. To see that $\sigma_f(T^*) \leq 3\sigma^*$, note that σ^* must be the distance between a voter and a candidate. Thus, there is one call of Algorithm 1 with $\sigma = \sigma^*$, which returns a committee $T \subseteq C$ of size at most k such that $\sigma_f(T) \leq 3\sigma = 3\sigma^*$, by Lemma 22. We have $\sigma_f(T^*) \leq \sigma_f(T)$ by construction, which implies $\sigma_f(T^*) \leq 3\sigma^*$.

Overall running time. We will show that each run of Algorithm 1 takes $O(nm^{2f+1})$ time. We can check the condition of while loop in Step 2 in time $O(m^{2f})$. This is because there are at most $O(m^f)$ failing sets of size at most f (the precise upper bound is $2m^f$), and for each failing set, we can find an optimal replacement in time $O(m^f)$ by bruteforce. Next, each iteration of the while loop takes $O(nm)$ time, that is, the time required to compute all voter-candidate pairwise distances. Therefore, the overall running time of the algorithm is $O(n^2m^{2f+2})$.

Thus, we have the following result.

Theorem 15 *We can find a 3-approximation for Optimal Fault-tolerant Committee in time $O(n^2m^{2f+2})$, assuming the fault-tolerance parameter f is a constant.*

Next, for a non-constant f, we give a 5 approximation using a greedy rule.

Lemma 23 *We can find a 5-approximation for Optimal Fault-Tolerant Committee in time $\mathcal{O}(mnk)$.*

Our algorithm is quite simple and uses the classical "farthest next" greedy rule [12]. Specifically, let C and V be the set of candidates and voters, respectively. We begin with an empty committee $T = \emptyset$ and an empty set $\hat{V}$ of picked voters. Then we repeat the following step: pick the voter $\hat{v} \in V \backslash \hat{V}$ farthest to the current committee T, add $\hat{v}$ to $\hat{V}$, and add to T the candidate $\hat{c} \in C$ closest to $\hat{v}$. The procedure terminates when $|T| = k$ or the candidate $\hat{c}$ computed is already in T. Formally, our algorithm is shown in Algorithm 2.

Algorithm 2 5-approximation algorithm for OFTC

Input: a set V of voters, a set C of candidates, the committee size k, and the fault-tolerance parameter f

$i \leftarrow 0$ and $T \leftarrow \emptyset$
while $|T| \leq k$ **do**
  $i \leftarrow i + 1$
  $v_i \leftarrow \arg\max_{v \in V \backslash \hat{V}} d(v, T)$
  $c_i \leftarrow \arg\min_{c \in C} d(v_i, c)$
  if $c_i \in T$ **then**
    break
  $T \leftarrow T \cup \{c_i\}$
return T

We now move on to the proof of correctness. Denote by σ^* the optimal f-tolerant score of a size-k committee. First, using the same analysis as the one for the k-center problem [46], we can show that $\sigma(T) \leq 3\sigma^*$.

Lemma 24 *Let T be the committee computed by Algorithm 2. Then $\sigma(T) \leq 3\sigma^*$.*

The proof Lemma 24 is easy and we refer the reader to [46] for details. We now show that T is a 5-approximate solution to OFTC.

Lemma 25 *Let T be the committee computed by Algorithm 2. Then $\sigma_f(T) \leq 5\sigma^*$.*

Proof: It suffices to show that for any failing set $J \subseteq C$ of size at most f, there exists a replacement set $K \subseteq C \backslash J$ such that $|K| = T \cap J$ and $\sigma_0((T \backslash J) \cup K) \leq 5\sigma^*$. Suppose $T = \{c_1, \ldots, c_r\}$, where c_i is the candidate selected in the i-th iteration of Algorithm 2. Let $v_1, \ldots, v_r$ be the voters computed in line 4 of Algorithm 2. For a failing set $J \subseteq C$, we construct the replacement set K as follows: For each index $i \in [r]$ such that $c_i \in J$, we include in K the candidate $c_i' \in C \backslash J$ closest to v_i. Clearly, $|K| = |T \cap J|$.

Now we show that $\sigma_0((T \backslash J) \cup K) \leq 5\sigma^*$. Using Lemma 24, we know that $d(v,T) \leq 3\sigma^*$ for any voter $v \in V$. Based on this, we bound $\sigma_0((T \backslash J) \cup K)$ as follows. Observe that $\sigma_0((T \backslash J) \cup K) = \max_{v \in V} d(v, (T \backslash J) \cup K)$. So it suffices to show that $d(v, (T \backslash J) \cup K) \leq 5\sigma^*$ for all $v \in V$. Let $c_i \in T$ be the candidate closest to v; thus, $d(v, c_i) = d(v,T) \leq 3\sigma^*$. If $c_i \notin J$, we are done. Otherwise, $c_i' \in K$ and hence $d(v, (T \backslash J) \cup K) \leq d(v, c_i')$. By the triangle inequality, we have

$$d(v, c_i') \leq d(v, c_i) + d(c_i, v_i) + d(v_i, c_i').$$

As argued before, $d(v, c_i) \leq 3\sigma^*$. Furthermore, $d(c_i, v_i) \leq d(v_i, c_i') \leq \sigma^*$, because c_i' is the candidate in $C \backslash J$ closest to v_i. Therefore, the above inequality implies $d(v, c_i') \leq 5\sigma^*$. ■

By the above lemma, we know that Algorithm 2 achieves an approximation ratio of 5. Its running time is clearly $O(mnk)$. This completes the proof of Lemma 23.

4.3 Bicriterion Approximation Scheme

In this final section of our chapter, we will present a bicriterion approximation scheme for OFTC. In particular, We give an EPTAS with running time $(1/\varepsilon)^{O(1/\varepsilon^{2d})}(m+n)^{O(1)}$ which is a *bicriterion* approximation, where the output committee T is fault-tolerant for at least $(1-\varepsilon)n$ voters with $\sigma_f(T) \leq (1+\varepsilon)\sigma^*$. Formally, we say a committee T is (r,ρ)-*good* if there exists a subset $V' \subseteq V$ of size at least ρn such that the f-tolerant score of T with respect to only the voters in V' is at most r. Then our approximation scheme can output a size-k committee which is $((1+\varepsilon)\sigma^*, 1-\varepsilon)$-good. The core of our approximation scheme is the following (approximation) decision algorithm. The decision algorithm takes the problem instance and an additional number $r > 0$ as input. The output of the algorithm has two possibilities: it either **(i)** returns YES and gives a size-k committee that is $((1+\varepsilon)r, 1-\varepsilon)$-good or **(ii)** simply returns NO. Importantly, the algorithm is guaranteed to give output **(i)** as long as $r \geq \sigma^*$. Note that this decision algorithm directly gives us the desired approximation scheme. Indeed, we can apply it with $r = d(v,c)$ for all $v \in V$ and $c \in C$. Let r^* be the smallest r that makes the algorithm give output **(i)**. The size-k committee T^* obtained when applying the algorithm with r^* is $((1+\varepsilon)r^*, 1-\varepsilon)$-good. We have $r^* \leq \sigma^*$ because the algorithm must be applied with $r = \sigma^*$ at some point and it is guaranteed to give output **(i)** at that time. Thus, T^* is $((1+\varepsilon)\sigma^*, 1-\varepsilon)$-good, as desired.

For simplicity of exposition, we describe our decision algorithm in two dimensions. By scaling, we may assume that the given number is $r = 1$. To solve the decision problem, our algorithm uses the shifting technique [17]. Let h be an integer parameter to be determined later. For a pair of integers $i, j \in \mathbb{Z}$, let $\square_{i,j}$ denote the $h \times h$ square $[i, i+h] \times [j, j+h]$. A square $\square_{i,j}$ is *nonempty* if it contains at least one voter or candidate. We first compute the index set $\widetilde{I} = \{(i,j) : \square_{i,j} \text{ is nonempty}\}$. This can be easily done in time $\mathcal{O}((n+m)h^2)$.

Consider a pair $(x,y) \in \{0,\ldots,h-1\}^2$. Let $L_{x,y}$ be the set of all integer pairs (i,j) such that $i \pmod h \equiv x$ and $j \pmod h \equiv y$. We write $\widetilde{I}_{x,y} = \widetilde{I} \cap L_{x,y}$. For a voter $v \in V$ and a square $\square_{i,j}$, we say v is a *boundary voter* for $\square_{i,j}$ if $v \notin [i+2, i+h-2] \times [j+2, j+h-2]$. Furthermore, we say v *conflicts* with (x,y) if v is a boundary voter in $\square_{i,j}$ for some

$(i, j) \in \widetilde{I}_{x,y}$.

Lemma 26 *There exists a pair $(x, y) \in \{0, \dots, h-1\}^2$ such that at most $\frac{4h-4}{h^2} \cdot |V|$ voters conflict with (x, y).*

Proof: A voter $v \in V$ may conflict with (x, y) only if for some $(i, j) \in \widetilde{I}_{x,y}$, we have $v \in \square_{i,j}$ but $v \notin [i+2, i+h-2] \times [j+2, j+h-2]$. Therefore, out of the total of h^2 pairs (x, y), v can only conflict with at most $h^2 - (h-2)^2$ pairs. Hence, using an averaging argument, there exists a pair (x, y) with at most $\frac{h^2-(h-2)^2}{h^2} \cdot |V|$ conflicting voters from V. ■

We fix a pair $(x, y) \in \{0, \dots, h-1\}^2$ that conflicts with the minimum number of voters. For $(i, j) \in \widetilde{I}_{x,y}$, we define the set of (non-boundary) voters $V_{i,j} = \{v \in \square_{i,j} : v \in [i+2, i+h-2] \times [j+2, j+h-2]\}$, and the set of candidates $C_{i,j} = \{c \in C : c \in \square_{i,j}\}$. Note that for $(i, j) \in \widetilde{I}_{x,y}$, the $C_{i,j}$'s are disjoint and form a partition of C. Next, we show an important lemma which allows our algorithm to divide our problem into smaller subproblems, solve them individually, and combine the solutions to solve the overall problem.

Lemma 27 *Let $V_1, V_2, \dots, V_s$ be subsets of V and let $T_1, T_2, \dots, T_s$ be pairwise disjoint subsets of C such that T_i is a fault-tolerant committee for V_i with $\sigma_f(T_i) = \sigma$. Then, $T = \bigcup_{i=1}^{s} T_i$ is a fault-tolerant committee of $\bigcup_{i=1}^{s} V_i$ with $\sigma_f(T) = \sigma$.*

Proof: We will show that for any failing set $J \subseteq C$, there exists a replacement set R with $|R| \leq |J \cap T|$ such that $\sigma_0((T \setminus J) \cup R) \leq \sigma$. For $i \in [s]$, let $J_i \subseteq J$ be the restriction of J to T_i, i.e., $J_i = J \cap T_i$. We know $|J| \leq f$. Since T_i is a fault-tolerant committee for V_i; hence, there exists a valid replacement $R_i \subseteq C \backslash J$ such that $|R_i| \leq |J_i|$ and $\sigma_0((T_i \backslash J_i) \cup R_i) \leq \sigma$. Let $R = \bigcup_{i=1}^{s} R_i$ (note that the R_i's need not be disjoint). Then we have $|R| \leq \sum_{i=1}^{s} |J_i| \leq |J \cap T_i|$ which implies $|R| \leq |J \cap T|$, and we have $\sigma_0((T \backslash J) \cup R) \leq \sigma$. This completes the proof of Lemma 27. ■

Consider a pair $(i,j) \in \widetilde{I}_{x,y}$. Let $\overline{T}_{i,j}$ be a smallest fault-tolerant committee for $V_{i,j}$ with $\sigma_f(\overline{T}_{i,j}) \leq 1$. We observe that any inclusion-minimal fault-tolerant committee $T_{i,j}$ for $V_{i,j}$ satisfies $T_{i,j} \subseteq C_{i,j}$. This is because any candidate outside $C_{i,j}$ has distance more than $1 + 6/h$ to any voter in $V_{i,j}$ (for a large enough value of h). In the next section we will show how to compute a fault-tolerant committee $T_{i,j} \subseteq C_{i,j}$ for $V_{i,j}$ such that $|T_{i,j}| \leq |\overline{T}_{i,j}|$ and $\sigma_f(T_{i,j}) \leq 1 + 6/h$ in $h^{O(h^4)} n^{O(1)}$ time. Assuming we can compute the above-mentioned committee $T_{i,j}$, our overall algorithm is as follows:

1. Fix a pair $(x,y) \in \{0, \ldots, h-1\}^2$ conflicting with the minimum number of voters, and set h to be the smallest integer such that $(4h-4)/h^2 \leq \varepsilon$ and $6/h \leq \varepsilon$.

2. For each pair $(i,j) \in \widetilde{I}_{x,y}$, compute $T_{i,j} \subseteq C_{i,j}$.

3. Let $T = \bigcup_{(i,j) \in \widetilde{I}_{x,y}} T_{i,j}$. If $|T| \leq k$, return YES (along with T); otherwise, return NO.

Let $V' = \bigcup_{(i,j) \in \widetilde{I}_{x,y}} V_{i,j}$. Since the $C_{i,j}$'s are disjoint, using Lemma 27, we conclude that T is a fault-tolerant committee for V'. Furthermore, from our choice of (x,y), we have $|V'| \geq (1-\varepsilon)n$. It is easy to show that the f-tolerant score of T with respect to the voters in V' is at most $1+\varepsilon$, and in addition, if $\sigma^* \geq 1$, we have $|T| \leq k$; we give a formal argument below. This proves correctness of our decision algorithm. The overall algorithm takes $(1/\varepsilon)^{O(1/\varepsilon^4)}(m+n)^{O(1)}$ time. We note that the algorithm can be directly generalized to the d-dimensional case with running time $(1/\varepsilon)^{O(1/\varepsilon^{2d})}(m+n)^{O(1)}$. Therefore, we have the following result.

Theorem 16 *Given a d-dimensional Fault-Tolerant Committee Selection instance, we can compute a size-k committee T such that the f-tolerant score of T with respect to at least $(1-\varepsilon)n$ voters is at most $(1+\varepsilon)\sigma^*$, where σ^* is the optimal f-tolerant score* *of a size-k committee (with respect to the entire set V). This algorithm runs in time* $(1/\varepsilon)^{O(1/\varepsilon^{2d})}(m+n)^{O(1)}$.

Correctness of the Decision Algorithm. Recall that, in our decision algorithm, we set h to be the smallest integer such that $(4h-4)/h^2 \leq \varepsilon$ and $6/h \leq \varepsilon$. Moreover, using Lemma 26 and our choice of (x, y), we have $|V'| \geq (1-\varepsilon)n$. If the computed committee T has size at most k, our algorithm returns YES; otherwise, it returns NO. To see the correctness our algorithm, recall that σ^* is an optimum score of a fault-tolerant committee for V. If $r \geq \sigma^*$, then there exists a size-k fault-tolerant committee with score r for V, and hence for V' (as $V' \subseteq V$).

Recall that for $(i,j) \in \tilde{I}_{x,y}$, the computed committee $T_{i,j}$ has $|T_{i,j}| \leq |\overline{T}_{i,j}|$ where $\overline{T}_{i,j}$ is the smallest committee for $V_{i,j}$ with $\sigma_f(\overline{T}_{i,j}) \leq 1$. Therefore, when $r \geq \sigma^*$, for $T = \bigcup_{(i,j)\in\tilde{I}_{x,y}} T_{i,j}$, we have $|T| \leq k$ and our algorithm returns YES. On the other hand, if there is no size-k committee whose f-tolerant score is at most $(1+\varepsilon)r$ for at least $(1-\varepsilon)n$ voters, we must have $|T| > k$ and thus our algorithm returns NO. This completes the argument for the proof of correctness.

4.3.1 Algorithm to Compute $T_{i,j}$

We now present the most challenging piece of our algorithm: the computation of the $T_{i,j}$'s. Consider a box $\square_{i,j}$. Suppose there exists a fault-tolerant committee $T \subseteq C$ for $V_{i,j}$ with $\sigma_f(T) \leq 1$. Our task is to compute a fault-tolerant committee $T_{i,j} \subseteq C$ for $V_{i,j}$ such that $|T_{i,j}| \leq |T|$ and $\sigma_f(T_{i,j}) \leq 1 + 6/h$.

We divide $\square_{i,j}$ into h^4 smaller cells each with size $\frac{1}{h} \times \frac{1}{h}$, and we denote the set of these cells by $L = \{l_1, \ldots, l_{h^4}\}$. (See Figure 4.3.) Our algorithm is based on two key observations:

(i) A committee with a candidate in every nonempty cell has f-tolerant score within a

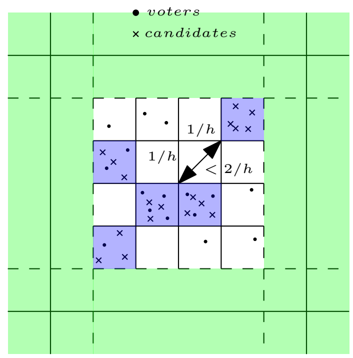


Figure 4.3: The figure shows a cell in the shifted grid. The solid lines around the sides are the grid lines (and the region inside them is a cell). The shaded (green) region is the boundary region. Inside the boundary region, we divide the cell into $1/h \times 1/h$ smaller cells. The distance between any two points in a smaller cell is $< 2/h$. All candidates in smaller cells are identical (i.e., candidates in blue regions). In this example, since only five cells are nonempty, we have at most 2^5 distinct failing sets.

difference of at most $2/h$ from the optimum score. Since the number of cells is h^4, this implies that the size of a smallest approximately optimal committee is bounded by h^4 (formally shown in Lemma 28).
(ii) All candidates in a cell can be treated as identical, causing only a loss of $2/h$ in the score. This implies that for any $T_{i,j}$, to approximately compute the f-tolerant score of $T_{i,j}$, we *only* need to consider the failing sets where either all or none of the candidates in a cell fail. Note that the number of such failing sets is at most $2^{O(h^4)}$ (formally shown in Lemma 29).

Using these two observations, at a high level, our algorithm goes through all committees of size at most h^4 (there are $h^{\mathcal{O}(h^4)}$ of these as we can assume that each cell has at most h^4 candidates), approximately computes the f-tolerant score of each of these committees in time $2^{O(h^4)}$, and returns the smallest one with the desired score.

Lemma 28 *Let $T, T^* \subseteq C$ be fault-tolerant committees for $V_{i,j}$. If $|T^* \cap l_a| = 1$ for all $a \in [h^4]$ such that $C \cap l_a \neq \emptyset$, then $\sigma_f(T^*) - \sigma_f(T) \leq 2/h$.*

Proof: Consider a failing set $J \subseteq C$. Since T is a fault-tolerant committee, there exists a valid replacement set R such that $|(T\backslash J) \cup R| \leq |T|$ and $\sigma_0((T\backslash J) \cup R) \leq \sigma_f(T)$. Let $L' = \{l_a \in L : l_a \cap ((T\backslash J) \cup R) \neq \emptyset\}$. Moreover, let $J^* = J \cap T^*$. Then we will show that there exists a replacement set R^* for J^* such that $|R^*| \leq |J^*|$ and $\sigma_0((T^* \setminus J^*) \cup R^*) - \sigma_0((T \setminus J) \cup R) \leq 2/h$.

We construct the set R^* as follows: Consider a cell $l_a \in L'$. Since $l_a \cap ((T \setminus J) \cup R) \neq \emptyset$, l_a is nonempty. This implies $T^* \cap l_a \neq \emptyset$ from the way we construct T^*. Let c_a be the only candidate in $T^* \cap l_a$. If $c_a \in J$, then we replace c_a with an arbitrary $c'_a \in l_a \setminus J$ (i.e., we add c'_a to R^*). We know that such a candidate c'_a exists because $l_a \cap ((T\backslash J) \cup R) \neq \emptyset$. Since we only add a candidate to R^* from l_a such that $J^* \cap l_a \neq \emptyset$, we have $|R^*| \leq |J^*|$.

We will now show that $\sigma_0((T^* \setminus J^*) \cup R^*) - \sigma_0((T \setminus J) \cup R) \leq 2/h$. Observe that for a pair of candidates $c_1, c_2 \in l_a$, $d(c_1, c_2) \leq 2/h$ (see Figure 4.3). Let $C_a = l_a \cap ((T \setminus J) \cup R)$. Moreover, let $V_a \subseteq V$ be the set of voters which have a candidate in C_a as their closest candidate in the committee $(T \setminus J) \cup R$. Using the triangle inequality, we know that $d(V_a, c'_a) \leq d(V_a, C_a) + 2/h$. The above statement holds for all cells $l_a \in L'$. Therefore, $\sigma_0((T^* \setminus J^*) \cup R^*) - \sigma_0((T \setminus J) \cup R) \leq 2/h$.

Note that our proof works for an arbitrary failing set J including $J = \emptyset$. This completes the proof of Lemma 28. ■

Based on the above observation, we solve the problem as follows. We enumerate all maps $\chi \colon L \to \{0, 1, \ldots, h^4\}$ where $\chi(l_a)$ is the number of candidates from l_a in the committee. The total number of such maps is $h^{O(h^4)}$. For each feasible map, i.e., χ satisfying $\chi(l_a) \leq |C \cap l_a|$ for all $a \in [h^4]$, we construct a fault-tolerant committee T^*_χ for $V_{i,j}$ by picking (arbitrarily) $\chi(l_a)$ candidates in $C \cap l_a$ for all $a \in [h^4]$ and including them in T^*_χ. For each constructed T^*_χ, we compute a number $\widetilde{\sigma_f}(T^*_\chi)$ that approximates $\sigma_f(T^*_\chi)$ using the following lemma.

Lemma 29 *Given T^*_χ, one can compute a number $\widetilde{\sigma_f}(T^*_\chi)$ in $2^{O(h^4)}n^{O(1)}$ time such that $|\widetilde{\sigma_f}(T^*_\chi) - \sigma_f(T^*_\chi)| \leq 2/h$.*

Proof: For a pair of candidates c_i, c_j in a cell $l_i \in L$, we know $d(c_i, c_j) \leq 2/h$. Since we want to compute the number $\tilde{\sigma_f}(T^*_\chi)$ within an absolute error of $2/h$ compared to the actual value, it is sufficient to only consider the failing sets for which either all or none of the candidates from a cell fail. The total number of cells is h^4; therefore, we only need to consider at most $2^{O(h^4)}$ distinct failing cells (see Figure 4.3). For each of these failing sets (say $J \subseteq C$), we can compute a best replacement committee R in time $2^{O(h^4)}$ by either choosing one or zero candidates from each cell. For each replacement, $\sigma_0(T^*_\chi \setminus J \cup R)$ can be computed in $O(nh^4)$ time. Therefore, we can compute $\sigma_f(\tilde{T^*_\chi})$ in time $2^{O(h^4)}n^{O(1)}$. ■

Finally, we let $T_{i,j}$ be the smallest among all committees T^*_χ satisfying $\widetilde{\sigma_f}(T^*_\chi) \leq 1 + 4/h$, and we return it as our solution. The running time of our algorithm is clearly $h^{O(h^4)}n^{O(1)}$. The following lemma shows that our algorithm is correct.

Lemma 30 *We have $\sigma_f(T_{i,j}) \leq 1 + 6/h$. Furthermore, $|T_{i,j}| \leq |T|$ for any fault-tolerant committee T for $V_{i,j}$ with $\sigma_f(T) \leq 1$.*

Proof: The fact that $\sigma_f(T_{i,j}) \leq 1 + 6/h$ follows directly from our construction and Lemma 29. Let T be a fault-tolerant committee for $V_{i,j}$ with $\sigma_f(T) \leq 1$. We consider two cases: $|T| > h^4$ and $|T| \leq h^4$. First, assume $|T| > h^4$. Define $\chi\colon L \to \{0, 1, \ldots, h^4\}$ by setting $\chi(l_a) = 1$ for all $a \in [h^4]$ with $C \cap l_a \neq \emptyset$, and $\chi(l_a) = 0$ whenever $C \cap l_a = \emptyset$. Clearly, $|T^*_\chi| \leq h^4 < |T|$. By Lemma 28, $\sigma(T^*_\chi) \leq 1 + 2/h$. Thus, $\widetilde{\sigma_f}(T^*_\chi) \leq 1 + 4/h$ by Lemma 29. This further implies that $|T_{i,j}| \leq |T^*_\chi| < |T|$.

Now assume $|T| \leq h^4$. Define $\chi\colon L \to \{0, 1, \ldots, h^4\}$ by setting $\chi(l_a) = |T \cap l_a|$ for all $a \in [h^4]$. Clearly, $|T^*_\chi| = |T|$ and $|T^*_\chi \cap l_a| = |T \cap l_a|$ for all $a \in [h^4]$. We show that $\sigma(T^*_\chi) \leq 1 + 2/h$. Since $|T^*_\chi \cap l_a| = |T \cap l_a|$, for each a pick a bijection $\pi_a\colon C \cap l_a \to C \cap l_a$ such that for all $x \in C \cap l_a$, $x \in T^*_\chi$ if and only if $\pi_a(x) \in T$. Observe that the distance between x and $\pi_a(x)$ is at most $2/h$ for all $x \in C \cap l_a$. Combining all bijections π_a, we obtain a bijection $\pi\colon C_{i,j} \to C_{i,j}$ with the property that for all $x \in C_{i,j}$, the distance between x and $\pi(x)$ is at most $2/h$, and $x \in T^*_\chi$ if and only if $\pi(x) \in T$. Because of this bijection, it is obvious that $|\sigma_f(T^*_\chi) - \sigma_f(T)| \leq 2/h$ and in particular $\sigma_f(T^*_\chi) \leq 1 + 2/h$. Thus, $\widetilde{\sigma_f}(T^*_\chi) \leq 1 + 4/h$ by Lemma 29. This further implies that $|T_{i,j}| \leq |T^*_\chi| = |T|$. ■

4.4 Bibliographic notes

To the best of our knowledge, the issue of fault tolerance in committee selection has not been studied in voting literature — their primary focus is on protocols and algorithms for choosing candidates [32, 33, 36, 37, 22, 47]. However, the following two lines of work consider some related issues. First, in the "unavailable candidate model" [48, 49] the goal is to choose a *single winner* with maximum expected score when candidates fail according to a given probability distribution; in contrast, we consider multiwinner elections under worst-case faults. In the second line of work, a set of election control problems are considered where candidates are added [50] or deleted [51] to change the outcome of the election. In this setting, the candidate set is modified to obtain a favorable election outcome, which is a rather different problem than ours.

In the facility-location research, there has been prior work on adding fault tolerance to k-center or k-median solutions [52, 53, 54, 55, 56], but the main approach there is to assign each user (voter) to multiple facilities (candidates). In particular, the "p-neighbor k-center" framework [52] minimizes the maximum distance between a user and its pth center as a way to protect against $p-1$ faults. This formulation, however, differs from our optimal fault-tolerant committee problem (OFTC) because in our setting the replacement candidates are chosen *after* failing candidates are announced. Therefore, in the OFTC problem, the designer does not have to *simultaneously* allocate p neighbors for all the voters. Furthermore, to the best of our knowledge, neither of our first two problems — Optimal Replacement (ORP) and Fault-Tolerance Score (FTS) — have been studied in the facility-location literature, and initiate a new research direction.

Chapter 5

Committee Winner Verification Problem

In the previous chapters, we studied the winner determination problem (Chapter 2) and the fault-tolerant committee selection (Chapter 3 and Chapter 4) under the Chamberlin-Courant voting rule for Euclidean elections. In this chapter, we take a step forward and focus our attention to the winner verification problems. In particular, given an election $E = (\mathcal{C}, V)$ (along with the preferences of voters over candidates), a desired committee size k and the voting rule under consideration we ask the following two questions:

1. Winner Verification: For a committee $S \subseteq \mathcal{C}$ of size at most k, is S a winning committee?

2. Candidate Winner: For a candidate $c \in \mathcal{C}$, does there exists a committee $S \subseteq \mathcal{C}$ of size at most k such that $c \in S$ and S is a winning committee?

Recall that a winning committee is the one with minimum (dissatisfaction/misrepresentation) score. The two questions above are natural to ask and answering them would be useful in many cases such as an election organizer might want check the individual/group fairness in terms of outcome possibilities of the election or a campaign manager might want to know if their party or a candidate has a chance of winning or not.

For an election with m voters, recall from Chapter 1 that Euclidean elections allow only a small fraction of all possible preference orderings over the candidates. (In Euclidean elections, the candidates and voters lie in a d-dimensional Euclidean space and the preferences are derived using the Euclidean distance.) In particular, only $\mathcal{O}(m^d)$ orderings are realizable in a d-dimensional Euclidean election among the total of all $m!$ orderings. In this chapter, we study *non-Euclidean elections*. That is, the candidates/voters are not embedded in a Euclidean space and the voter preferences need not follow any structures (such as the one imposed by Euclidean distances). Note that all $m!$ orderings of the candidates are realizable in this scenario. We work with the Chamberlin-Courant [11] and Monroe [18] voting rules. Recall from Chapter 1 that both these rules ensure representation as each voter is "assigned" to a committee member. Furthermore, the Monroe voting rule also ensure proportional representation as each committee member is assigned equal number of voters.

We recall briefly that the preferences of voters in an election instance are typically solicited as either *rankings* (total orders over candidates) or *approval ballots* (subsets of "approved" candidates) – see Chapter 1 for more details. The problem of finding a committee whose misrepresentation is bounded by a given threshold is known to be NP-complete for Chamberlin-Courant and Monroe [57, 58] in the setting of rankings as well as approval ballots. In a recent development ([59, Theorem 10], improving upon [60, Corollary 3]), it was shown that it is Θ_2^P-hard to determine whether a given candidate belongs to an optimal CC committee in the setting of rankings for the utilitarian method of aggregating misrepresentation scores. Following up on this, the main contribution of our work is to completely settle the complexity of two aforementioned natural versions of the winner determination question in the context of the two fundamental multiwinner rules — Chamberlin-Courant and Monroe. We address these problems in the settings of both rankings and approval ballots, and for both the utilitarian and egalitarian methods of aggregating scores.

5.0.1 Results and Organization of the Chapter

Our first set of contributions is for the WINNER VERIFICATION problem; we show that it is complete for the complexity class coNP. In this case, the membership is easy to establish. For a given committee, observe that it is easy to compute its score with respect to the Chamberlin-Courant rule (and also the Monroe rule, although this is less straightforward). Thus, our coNP certificate is simply a "rival" committee with a better score. We remark, as an aside, that this is in contrast with rules such as Dodgson [61, 62] for which computing the Dodgson score of a given candidate is intractable.

To show hardness for coNP, we reduce from the complement of the HITTING SET problem in different ways depending on the setting. We present the results for Chamberlin-Courant rule under ranking preferences in Section 5.1. For showing the hardness of the *Monroe Rule* we employ a variant where the elements enjoy uniform occurrences among the sets (See Section 5.2 for details). We consider the case of approval voting in Section 5.3 and we show show that WINNER VERIFICATION stays coNP-complete even in this case for both Chamberlin-Courant and Monroe voting rule. Apart from settling the complexity of fundamental question of winner verification, our contribution identifies a natural coNP-complete problem, in particular, one that is *not* merely the complement of a natural NP-complete problem. Finally, in Section 5.4, we give polynomial-time algorithms for the winner verification problem under single-peaked or single-crossing preferences for both the Chamberlin-Courant voting rule.

For the ease of presentation, we present our results on the CANDIDATE WINNER problem in Chapter 6.

5.0.2 Additional notations for this Chapter

For a positive integer ℓ, we denote the set $\{1, \ldots, \ell\}$ by $[\ell]$. For convenience, for the rest of this chapter and for Chapter 6 we use C for denote the set of candidates instead of $\mathcal{C}$. Every voter v has a *preference* $\succ_v$ which is typically a complete order over the set C of candidates (rankings) or a subset of approved candidates (approval ballots). An instance of an election consists of the set of candidates C and the preferences of the voters V, usually denoted as $E = (C, V)$ with the understanding that the voters in V are identified by their preferences.

We now recall some definitions in the context of rankings. We say voter v prefers a candidate $x \in C$ over another candidate $y \in C$ if $x \succ_v y$. For a ranking $\succ$, $\mathrm{pos}_{\succ}(c)$ is given by one plus the number of candidates ranked above c in $\succ$. In particular, if there are m candidates and c is the top-ranked (respectively, bottom-ranked) candidate in the ranking $\succ$, then $\mathrm{pos}_{\succ}(c)$ is one (respectively, m). We denote the set of all preferences over C by $\mathcal{L}(C)$. The n-tuple $(\succ_v)_{v \in V} \in \mathcal{L}(C)^n$ of the preferences of all the voters is called a *profile*. We note that a profile, in general, is a multiset of linear orders. For a subset $M \subseteq V$, we call $(\succ_v)_{v \in M}$ a sub-profile of $(\succ_v)_{v \in V}$. For a subset of candidates $D \subseteq C$, we use $\mathcal{P}|_D$ to denote the projection of the profile $\mathcal{P}$ on the candidates in D alone. The definitions of profiles, sub-profiles, and projections are analogous for approval ballots.

Chamberlin-Courant for Rankings. The Chamberlin–Courant voting rule is based on the notion of a *dissatisfaction* or a *misrepresentation* function. This function specifies, for each $i \in [m]$, a voter's dissatisfaction $\alpha^m(i)$ from being represented by the candidate she ranks in position i. A popular dissatisfaction function is Borda, given by $\alpha^m(i) = i - 1$.

We now turn to the notion of an assignment function. Let $k \leq m$ be a positive integer. A *k-CC-assignment function* for an election $E = (C, V)$ is a mapping $\Phi \colon V \to C$ such that $|\Phi(V)| = k$, where $\Phi(V)$ denotes the image of Φ. For a given assignment function Φ, we say that voter $v \in V$ is *represented* by candidate $\Phi(v)$ in the chosen committee. There are several ways to measure the quality of an assignment function Φ with respect to a dissatisfaction function $\alpha : [m] \longrightarrow \mathbb{R}$; and we will use the following:

1. $\ell_1(\Phi, \alpha) = \sum_{v \in V} \alpha(\mathrm{pos}_{\succ_v}(\Phi(v)))$, and

2. $\ell_\infty(\Phi, \alpha) = \max_{v \in V} \alpha(\mathrm{pos}_{\succ_v}(\Phi(v)))$.

Unless specified otherwise, α will be the Borda dissatisfaction function described above. We are now ready to define the Chamberlin-Courant voting rule.

Definition 1 (Chamberlin-Courant) *For $\ell \in \{\ell_1, \ell_\infty\}$, the* ℓ-CC voting rule *is a mapping that takes an election $E = (C, V)$ and a positive integer k with $k \leq |C|$ as its input, and returns the images of all the k-CC-assignment functions Φ for E that minimizes $\ell(\Phi, \alpha)$.*

Chamberlin-Courant for Approval Ballots. Recall that an approval vote of a voter v on the set of candidates C is some subset S_v of C such that v approves all the candidates in S_v. We define the misrepresentation score of a k-sized committee W as the number of voters which do not have any of their approved candidates in W (i.e. $W \cap S_v = \emptyset$). Hence the optimal committees under approval Chamberlin-Courant are the committees which maximize the number of voters with at least one approved candidate in the winning committee [63].

We now turn to the definition of the Monroe voting rule [18]. Note that for $c \in C$, $\Phi^{-1}(c)$ denotes the set of voters *represented* by c.

Definition 2 (Monroe) *For $\ell \in \{\ell_1, \ell_\infty\}$, the* ℓ-Monroe voting rule *is a mapping that takes an election $E = (C, V)$ and a positive integer k with $k \leq |C|$ as its input, and returns the image of any of the k-Monroe-assignment functions Φ such that $|\Phi^{-1}(c)|$ is either $\lfloor \frac{n}{k} \rfloor$ or $\lceil \frac{n}{k} \rceil$ where $c \in C$ for E that minimizes $\ell(\Phi, \alpha)$.*

We are now ready to describe the questions that we study in this chapter. The first problem is Chamberlin-Courant Winner Verification **(CCWV)**. Here, the input is an election $E = (C, V)$ and a subset S of k candidates. The question is if S is a winning k-sized CC-committee for the election E, in other words, does S achieve the best Chamberlin-Courant score in the given election among all committees of size k? The second problem is Monroe Winner Verification **(MWV)** which is defined in a similar way.

We also recall the definitions of 3-Hitting Set and its *complement* as we will use these problems in our reductions. In the 3-Hitting Set problem, we are given a ground set $\mathcal{U}$, a family $\mathcal{F}$ of three-sized subsets of $\mathcal{U}$, and an integer k, and the question is if there exists $S \subseteq \mathcal{U}$ of size at most k that intersects every set in $\mathcal{F}$, i.e: $\forall\ F \in \mathcal{F}$, $S \cap F \neq \phi$. In the c-3-Hitting Set problem, the input is the same, and is a Yes-instance if and only if there is no hitting set of size k; in other words, if for each $S \subseteq \mathcal{U}$ with $|S| \leq k$, there exists some $F_S \in \mathcal{F}$ such that $S \cap F_S = \phi$. We recall that 3-Hitting Set is a classic NP-complete problem, and c-3-Hitting Set is *coNP*-complete.

The Class $P_{\parallel}^{NP}$ (Θ_2^P). The class $P_{\parallel}^{NP}$ is the class of problems solvable using a P machine having parallel access to an NP oracle. The class Θ_2^P was introduced in [64] and named in [65]. The class Θ_2^P was shown to be equivalent to $P_{\parallel}^{NP}$ by Hemachandra [66]. The Vertex Cover Member problem is the following. Given a graph $G := (V, E)$ and a vertex $w \in V$, the question is if there exists a minimum sized vertex cover containing w. The problem was shown to be complete for $P_{\parallel}^{NP}$ by [67].

Restricted Preference Domains. In this work, we consider two well-studied notions of restricted preference domains, namely, single-peaked and single-crossing preferences domains.

Single-Peaked Preference Domains. A preference profile is said be single-peaked if there exists an ordering σ over the candidates C such that the preference of every voter v has the following structure: v has a favorite candidate c (sometimes called the "peak" for v), and the further away a candidate $d \neq c$ is from c in σ, the less it is preferred by the voter v. A formal definition is as follows:

Definition 3 (Single-Peaked Domain) *A preference $\succ \in \mathcal{L}(C)$ over a set of candidates C is called* single-peaked *with respect to an order $\succ' \in \mathcal{L}(C)$ if, for every pair of candidates $x, y \in C$, we have $x \succ y$ whenever we have either $c \succ' x \succ' y$ or $y \succ' x \succ' c$, where $c \in C$ is the candidate at the first position of $\succ$. A profile $\mathcal{P} = (\succ_i)_{i \in [n]}$ is called single-peaked with respect to an order $\succ' \in \mathcal{L}(C)$ if $\succ_i$ is single-peaked with respect to $\succ'$ for every $i \in [n]$.*

Single-Crossing Preference Domains. A preference profile is said to belong to the single-crossing domain if it admits a permutation of the voters such that for any pair of candidates a and b, there is an index $j\langle a, b\rangle$ such that either all voters v_j with $j < j\langle a, b\rangle$ prefer a over b and all voters v_j with $j > j\langle a, b\rangle$ prefer b over a, or vice versa. The formal definition is as follows.

Definition 4 (Single-Crossing Domain) *A profile $\mathcal{P} = (\succ_i)_{i \in [n]}$ of n preferences over a set C of candidates is called a* single-crossing profile *if there exists a permutation σ of $[n]$ such that, for every pair of distinct candidates $x, y \in C$, whenever we have $x \succ_{\sigma(i)} y$ and $x \succ_{\sigma(j)} y$ for two integers i and j with $1 \leq \sigma(i) < \sigma(j) \leq n$, we have $x \succ_{\sigma(k)} y$ for every $\sigma(i) \leq k \leq \sigma(j)$.*

5.1 Chamberlin-Courant Voting Rule

In this section, we show the coNP-completeness of Chamberlin-Courant Winner Verification in the setting of rankings for the ℓ_1-Borda misrepresentation score. The argument for membership is, in brief, the following: a rival committee with a better misrepresentation score is a valid certificate for the No instances of CCWV. This is an efficiently computable certificate since it is easy to compute the Chamberlin-Courant score of a given committee. We now turn to the proof of hardness.

Theorem 17 Chamberlin-Courant Winner Verification *is coNP-hard in the setting of rankings for the ℓ_1-Borda misrepresentation score.*

Proof: We show a reduction from c-3-Hitting Set to the CCWV problem. Let $\langle U, \mathcal{F}; k \rangle$ be an instance of c-3-Hitting Set with n elements in the universe U and m sets of size three in the family $\mathcal{F}$. We construct a profile $\mathcal{P}$ over alternatives $\mathcal{A}$ as follows. First, we introduce one candidate corresponding to each element of the universe U, k "dummy" candidates, and a large number of "filler" candidates, that is:

$$\mathcal{A} := \underbrace{\{c_u \mid u \in U\}}_{\mathcal{C}} \ \cup \ \underbrace{\{d_1, \ldots, d_k\}}_{\mathcal{D}} \ \cup \ \underbrace{\{z_1, \ldots, z_t\}}_{Z},$$

where $t = 3(mk)^2$. Also, for every $1 \leq i \leq k$, and for every $X \in \mathcal{F}$, introduce a vote $v(i, X)$ that places the candidates corresponding to the elements in X in the top three positions, followed by d_i, followed by $3mk$ candidates from Z. We ensure that we use distinct candidates from Z in the top $3mk + 4$ positions of all the voters, in other words, no candidate from Z appears twice in the top $3mk + 4$ positions. Note that t is chosen to be large enough to make this possible. This is followed by the candidates in $U \setminus X$ ranked in an arbitrary order followed by the remaining filler candidates which are also ranked in an arbitrary order.

In this instance, note that a committee corresponding to a hitting set has a score of at most $2mk$, while the score of the committee $\mathcal{D}$ is $3mk$. In the constructed instance, we now ask if the committee $\mathcal{D}$ consisting of k dummy candidates is a winning committee. This completes the construction of the instance. We now turn to the equivalence of two instances.

In the forward direction, suppose we have a YES instance of c-3-HITTING SET then this implies that there does not exist any hitting set of size at most k. Recall that the misrepresentation score for a committee consisting of a hitting set is at most $2mk$, while noting that any such committee must have size greater than k. Now, we show that for all other committees of size at most k, the misrepresentation score is greater than $3mk$.

Lemma 31 *Consider an instance $\langle \mathcal{A}, V, \mathcal{D} \rangle$ of CC-winner Verification based on a* YES-*instance of c-3-Hitting Set $\langle U, \mathcal{F}; k \rangle$. For any feasible committee $C' \subseteq \mathcal{A}$ of size k different from $\mathcal{D}$, the ℓ_1-Borda misrepresentation score of C' is greater than $3mk$.*

Proof: Let U', D' and Z' denote, respectively, the candidate subsets $C' \cap \mathcal{C}$, $C' \cap \mathcal{D}$ and $C' \cap Z$. Since C' is different from $\mathcal{D}$, there is at least one candidate from $\mathcal{D}$ that does not belong to C' (the only other possibility is that C' is a superset of $\mathcal{D}$, but this is not possible since $|C'| = |\mathcal{D}| = k$). Without loss of generality, suppose $d_1 \notin C'$. Now consider the votes given by $V' := \{v(1, X) \mid X \in \mathcal{F}\}$. We claim that there are at least $|Z'| + 1$ voters in V' whose misrepresentation score for the committee C' is strictly greater than three. Indeed, if not, then it is straightforward to verify that U' combined with an arbitrarily chosen element from each set not hit by U' comprises a subset of size at most $|U'| + |Z'| \leq k$ which intersects every set in $\mathcal{F}$, contradicting our assumption that $\mathcal{F}$ has no hitting set of size at most k. To see this, observe that every vote in V' that has a misrepresentation score of three or less is necessarily represented by a candidate from U', since $d_1 \notin C'$, and therefore, the sets corresponding to all of these votes are hit by U', and the remaining sets can be hit "trivially" since there are at most $|Z'|$ of them. Now consider the voters who have a "high" misrepresentation score: $V'' := \{v(1, X) \mid X \in \mathcal{F}$ and $\tau(v(1, X), C') > 3\}$, where $\tau(v(1, X), C')$ is the Borda score of the highest-ranked candidate of C' according to $v(1, X)$, with respect to the ranking of $v(1, X)$. By the argument in the previous paragraph, we have that $|V''| > |Z'|$. Recalling that every vote has distinct filler candidates in the top $3mk$ positions after d_i, by the pigeon-hole principle, we conclude that there is at least one vote $v(1, X)$ in V'' such that $Z_X \cap Z' = \emptyset$, where Z_X denotes the filler candidates that appear in the top $3mk + 4$ positions of the vote $v(1, X)$. Since the candidates occupying the top four positions of this vote do not belong to C' either, it follows that the misrepresentation score of $v(1, X)$ for C' is greater than $3mk$, and this concludes our argument. ■

The committee $\mathcal{D}$ has a misrepresentation score of $3mk$. Using Lemma 31, since $\mathcal{F}$ has no hitting set of size at most k, we have that $\mathcal{D}$ is a winning committee among all feasible committees, as desired.

In the reverse direction, we start with the assumption that $\mathcal{D}$ is a winning committee. Therefore, the optimal misrepresentation for the constructed election instance is $3mk$. Observe that if there exists a hitting set S of size at most k, then the committee C' formed using the corresponding candidates of hitting set will have a misrepresentation score of at most $2mk$, as discussed above. Thus, $\mathcal{D}$ would not be a winning committee, a contradiction — and this implies that $\langle U, \mathcal{F}; k\rangle$ was indeed a YES-instance of c-3-HITTING SET. This completes the argument of equivalence. ■

Next, we show an analogous result for ℓ_∞-Borda misrepresentation score. We note that our arguments are similar to the previous result, although the construction is simpler in this case.

Theorem 18 CHAMBERLIN-COURANT WINNER VERIFICATION *is coNP-hard in the* *setting of rankings for the ℓ_∞-Borda misrepresentation score.*

Proof: We reduce from c-3-Hitting Set to our problem. Let $\langle U, \mathcal{F}; k\rangle$ be an instance of c-3-Hitting Set with n elements in the universe U and m sets of size three in the family $\mathcal{F}$. We construct a profile $\mathcal{P}$ over alternatives $\mathcal{A}$ as follows. We introduce one candidate corresponding to each element of the universe U, and k "dummy" candidates, that is:

$$\mathcal{A} := \underbrace{\{c_u \mid u \in U\}}_{\mathcal{C}} \cup \underbrace{\{d_1, \ldots, d_k\}}_{\mathcal{D}}$$

Also, for every $1 \leq i \leq k$, and for every $X \in \mathcal{F}$, introduce a vote $v(i, X)$ that places the candidates corresponding to the elements in X in the top three positions, followed by d_i, followed by the candidates in $U \setminus X$ ranked in an arbitrary order.

In this instance, note that a committee corresponding to a hitting set has a score of at most 2, while the score of the committee $\mathcal{D}$ is 3. In the constructed instance, we now ask if the committee $\mathcal{D}$ consisting of k dummy candidates is a winning committee. This completes the construction of the instance. We now turn to the equivalence of two instances.

In the forward direction, suppose we have a Yes instance of c-3-Hitting Set. This implies that there does not exist any hitting set of size at most k. Recall that the misrepresentation score for a committee consisting of a hitting set is at most 2, while noting that any such committee must have size greater than k. We show that the misrepresentation score of any k-sized committee other than $\mathcal{D}$ has a strictly greater misrepresentation score than 3.

Lemma 32 *For a constructed* CCWV *instance based on* Yes*-instance of* c-3-Hitting Set*, for any feasible committee $C' \subseteq A$ of size k different from $\mathcal{D}$, the ℓ_∞-Borda misrepresentation score of C' is greater than* 3.

Proof: Let U', and D' denote, respectively, the candidate subsets $C' \cap \mathcal{C}$, and $C' \cap \mathcal{D}$. Since C' is different from $\mathcal{D}$, and $|C'| = |\mathcal{D}| = k$, there is at least one candidate from $\mathcal{D}$ that does not belong to C'. Without loss of generality, suppose $d_1 \notin C'$. Now consider the votes given by $V' := \{v(1, X) \mid X \in \mathcal{F}\}$. We claim that at least one voter in V' whose misrepresentation score for the committee C' is strictly greater than three. Indeed, if not, then given that $d_1 \notin C'$, it is straightforward to verify that U' is a subset candidates of size at most k which intersect every set in $\mathcal{F}$, contradicting our assumption that $\mathcal{F}$ has no hitting set of size at most k. To see this, observe that every vote in V' that has a misrepresentation score of three or less is necessarily represented by a candidate from U', since $d_1 \notin C'$, and therefore, the sets corresponding to all of these votes are hit by U'. This completes the argument for Lemma 32. ■

Note that the committee with k dummy candidates $\mathcal{D}$ has misrepresentation score 3. Using Lemma 32, since $\mathcal{F}$ has no hitting set of size at most k, we have that $\mathcal{D}$ is a winning committee among all feasible committees, as desired.

For the reverse direction, we are given that $\mathcal{D}$ is a winning committee for the constructed CCWV instance. Hence, the optimal misrepresentation score is 3. It is easy to see that if there exists a hitting set S of size at most k, then the committee C' formed using the corresponding hitting set candidates is at most 2. Thus, if there exists a hitting set, $\mathcal{D}$ is not a winning committee – this contradicts the case we are in. Therefore, $\langle U, \mathcal{F}; k \rangle$ was a YES-instance of c-3-HITTING SET. ■

5.2 Monroe Voting Rule

In this section, we show an analogous set of results for the case of Monroe Voting rule. Specifically, we will show that MONROE WINNER VERIFICATION problem is coNP-complete for both ℓ_1, ℓ_∞ Borda-misrepresentation functions.

First, we briefly describe the argument for membership to coNP: We recall that, given a committee S, one can compute its Monroe misrepresentation score in polynomial time [58, 68]. Note that this is not as straightforward as evaluating the misrepresentation score of a Chamberlin-Courant committee due to the strict assignment conditions for the Monroe rule. However, the problem admits an efficient algorithm by a reduction to the minimum cost maximum flow problem. Let the S be a part of input for MONROE WINNER VERIFICATION problem. Furthermore, let the misrepresentation score of S be r. Consider a committee C' of at most k candidates with misrepresentation r' such that $r' < r$. Notice that C' is a valid efficient *certificate* for NO-instance of our problem (because we can compute r' in polynomial time using by computing the minimum cost maximum flow efficiently). Hence, by comparing the misrepresentation scores of committee S and the *guessed* rival committee C' under Monroe voting rule, we can efficiently verify the NO-instances of the problem. Therefore, We conclude that MONROE WINNER VERIFICATION is in coNP.

Towards establishing the coNP-hardness of MONROE WINNER VERIFICATION, we reduce from a special version of HITTING SET, which we refer to as BOUNDED OCCURRENCE EXACT HITTING SET (BOEHS for brevity). Here, the instance is $\langle U, \mathcal{F}; k\rangle$, where U is a finite universe with $3q$ elements, $\mathcal{F}$ is a collection of three-sized subsets of U such that every element of U appears in exactly three sets, and the question is if there exists a subset S of size $k = q$ that hits every set. Note that $|U| = 3q$ and $|S| = q$ combined with the frequency assumption imply that if S is a solution, then every set is hit exactly once.

The NP-hardness of BOEHS follows from the duality of Hitting Set and Set Cover. In particular, consider the problem of X3C with bounded occurrences [12]. In this problem, we are given a collection of three-sized subsets of an universe U with $3q$ elements where every element occurs in exactly three sets, and the question is if there is a collection of

q subsets whose union is U. Note that these sets must necessarily be disjoint. Given an instance $\langle U, \mathcal{F}; q\rangle$ of X3C with bounded occurrences, we obtain an instance of BOEHS by taking the dual (note that $|\mathcal{F}| = |U| = 3q$):

$$U' = \{u_X \mid X \in \mathcal{F}\}; \mathcal{F} = \{Y_u \mid u \in U\} \text{ where } Y_u = \{u_X \mid u \in X\}; k = q.$$

The equivalence of these instances is easy to verify and implies the NP-hardness of BOEHS. For our reductions, we reduce from c-BOEHS, which is the complement problem and is coNP-complete.

Theorem 19 Monroe Winner Verification *is coNP-hard in the setting of rankings for the ℓ_1-Borda misrepresentation score.*

Proof: We note that our construction is similar to the one used in Theorem 17. We show a reduction from c-BOEHS to our problem. Let $\langle U, \mathcal{F}; k\rangle$ be an instance of c-BOEHS with $3q$ elements in the universe U and $3q$ sets of size three in the family $\mathcal{F}$ (note that here $k = q$). We construct a profile $\mathcal{P}$ over alternatives $\mathcal{A}$ as follows. First, we introduce one candidate corresponding to each element of the universe U, k "dummy" candidates, and a large number of "filler" candidates, that is:

$$\mathcal{A} := \underbrace{\{c_u \mid u \in U\}}_{\mathcal{C}} \cup \underbrace{\{d_1, \ldots, d_k\}}_{\mathcal{D}} \cup \underbrace{\{z_1, \ldots, z_t\}}_{Z},$$

where $t = 3(3qk)^2$. Also, for every $1 \leq i \leq k$, and for every $X \in \mathcal{F}$, introduce a vote $v(i, X)$ that places the candidates corresponding to the elements in X in the top three positions, followed by d_i, followed by $9qk$ candidates from Z. We ensure that we use distinct candidates from Z in the top $9qk + 4$ positions of all the voters, in other words, no candidate from Z appears twice in the top $9qk+4$ positions. Note that t is chosen to be large enough to make this possible. This is followed by the candidates in $U \setminus X$ ranked in an arbitrary order followed by the remaining filler candidates, also ranked in an arbitrary order.

In the constructed instance, consider a committee $S' \subseteq C$ of size k corresponding to a Bounded Occurrence Exact Hitting Set. For each $s \in S'$, we assign exactly $3k$ votes to s. In particular, for each $i \in [k]$, s represents exactly 3 votes $v(i, X_j)$ for $j \in [3]$ such that set X_j contains the element corresponding to candidate s. Note that such a committee S has the dissatisfaction score of at most $6qk$. Consider the committee $\mathcal{D}$. Here, each $d_i \in \mathcal{D}$ represents $3q$ voters $v(i, X)$ for $X \in \mathcal{F}$. The misrepresentation score of the committee $\mathcal{D}$ is $9qk$. In the constructed instance, we now ask if the committee $\mathcal{D}$ consisting of k dummy candidates is a winning committee. This completes the construction of the instance. We now turn to the equivalence of two instances.

In the forward direction, suppose we have a Yes instance of c-BOEHS. This implies that there does not exist any hitting set of size at most k. Recall that the misrepresentation score for a committee consisting of a hitting set is at most $6qk$, while noting that any such committee must have size greater than k. Now, we show that for all other committees of size at most k, the misrepresentation score is greater than $9qk$.

Lemma 33 *Consider an instance $\langle \mathcal{A}, V, \mathcal{D} \rangle$ of Monroe-winner Verification based on a* Yes*-instance of c-*BOEHS *$\langle U, \mathcal{F}; k \rangle$. For any feasible committee $C' \subseteq \mathcal{A}$ of size k different from $\mathcal{D}$, the ℓ_1-Borda misrepresentation score of C' is greater than $9qk$.*

Proof: Let U', D' and Z' denote, respectively, the candidate subsets $C' \cap \mathcal{C}$, $C' \cap \mathcal{D}$ and $C' \cap Z$. Since C' is different from $\mathcal{D}$, there is at least one candidate from $\mathcal{D}$ that does not belong to C' (the only other possibility is that C' is a superset of $\mathcal{D}$, but this is not possible since $|C'| = |\mathcal{D}| = k$). Without loss of generality, suppose $d_1 \notin C'$.

Now consider the votes given by $V' := \{v(1, X) \mid X \in \mathcal{F}\}$. We claim that there are at least $|Z'| + 1$ voters in V' whose misrepresentation score for the committee C' is strictly greater than three. Indeed, if not, then it is straightforward to verify that U' combined with an arbitrarily chosen element from each set not hit by U' comprises a subset of size at most $|U'| + |Z'| \leq k$ which intersects every set in $\mathcal{F}$, contradicting our assumption that $\mathcal{F}$ has no hitting set of size at most k. To see this, observe that every vote in V' that has a misrepresentation score of three or less is necessarily represented by a candidate from U', since $d_1 \notin C'$, and therefore, the sets corresponding to all of these votes are hit by U', and the remaining sets can be hit "trivially" since there are at most $|Z'|$ of them. Now consider the voters who have a "high" misrepresentation score: $V'' := \{v(1, X) \mid X \in \mathcal{F} \text{ and } \tau(v(1, X), C') > 3\}$, where $\tau(v(1, X), C')$ is the Borda score of the highest-ranked candidate of C' according to $v(1, X)$, with respect to the ranking of $v(1, X)$. By the argument in the previous paragraph, we have that $|V''| > |Z'|$. Recalling that every vote has distinct filler candidates in the top $9qk$ positions after d_i, by the pigeon-hole principle, we conclude that there is at least one vote $v(1, X)$ in V'' such that $Z_X \cap Z' = \emptyset$, where Z_X denotes the filler candidates that appear in the top $9qk + 4$ positions of the vote $v(1, X)$. Since the candidates occupying the top four positions of this vote do not belong to C' either, it follows that the misrepresentation score of $v(1, X)$ for C' is greater than $9qk$, and this concludes our argument. ■

Recall that the misrepresentation for $\mathcal{D}$ is $9qk$. Hence, using Lemma 33, it is easy to see that $\mathcal{D}$ is a winning committee for elections constructed from the YES instance of c-BOEHS.

In the reverse direction, we assume $\mathcal{D}$ is a winning committee for the constructed election instance. Hence, the optimal misrepresentation score is $9qk$. Notice that if there exists a hitting set of size at most k, then the misrepresentation of the committee C' formed using the candidates corresponding to the elements in the hitting set is at most

$6qk$. Hence, if there exists a hitting set, $\mathcal{D}$ is not a winning committee – this contradicts the case we are in. Therefore, $\langle U, \mathcal{F}; k\rangle$ was a Yes-instance of c-BOEHS.

■

The Monroe Winner Verification problem is coNP complete for ℓ_∞-Borda misrepresentation score in the setting of rankings. The argument for membership is similar to the one described at the beginning of this section, and to show the hardness, we again reduce from c-BOEHS. Due to similarity of arguments with Theorem 18, we omit the details.

5.3 Elections with Approval Preferences

In this section, we turn to the approval ballots setting. We begin with the Chamberlin-Courant voting rule. A membership argument described at the start of Section 5.1 works for the case of approval ballots as well. Although we again reduce from the c-3-Hitting Set problem, the arguments have to account for a non-positional misrepresentation function.

Theorem 20 Chamberlin-Courant Winner Verification *is coNP-hard even when the preferences are presented as approval ballots for the ℓ_1-Borda misrepresentation score.*

Proof: Let $\langle U, \mathcal{F}; k\rangle$ be an instance of c-3-Hitting Set. Recall that this is a Yes-instance if and only if for all $S \subseteq U$, with $|S| \leq k$, there exists $X \in \mathcal{F}$ such that $S \cap X = \emptyset$. We construct a profile $\mathcal{P}$ over alternatives $\mathcal{A}$ as follows. First, we introduce one candidate corresponding to each element of the universe U, and k "dummy" candidates, that is:

$$\mathcal{A} := \underbrace{\{c_u \mid u \in U\}}_{\mathcal{C}} \ \cup \ \underbrace{\{d_1, \ldots, d_k\}}_{\mathcal{D}}$$

Also, for every $1 \leq i \leq k-1$, and for every $X \in \mathcal{F}$, introduce a vote $v(X, i)$ that approves the candidates corresponding to the elements in X along with d_i. For $i = k$, we introduce the same set of voters with the same candidate approval scheme except for one arbitrarily chosen voter which does not approve d_k.

We note that in the constructed instance, the score of a committee corresponding to a hitting set is zero whilst the score of committee with k dummy candidates $\mathcal{D}$ is one. In this instance, we ask if the committee with k-dummy candidates is a winning committee.

In the forward direction, we have a 'YES' instance of c-3-Hitting Set. Hence, $\langle U, \mathcal{F}; k\rangle$ does not admit hitting set of size at most k. Recall that the misrepresentation score of the committee corresponding to a hitting set is zero, but in this case, any such committee will have size strictly greater than k. Now, we show that for all other committees, there exists at least one voter, which contributes positively to misrepresentation; hence, their misrepresentation score is at least one.

Lemma 34 *For a constructed* CCWV *instance based on* Yes*-instance of* c-3-Hitting Set*, for any feasible committee $C' \subseteq A$ of size k different from $\mathcal{D}$, the ℓ_1 misrepresentation score of C' is strictly positive.*

Proof: Let U', and D' denote, respectively, the candidate subsets $C' \cap \mathcal{C}$, and $C' \cap \mathcal{D}$. Since C' is different from $\mathcal{D}$, and $|C'| = |\mathcal{D}| = k$, there is at least one candidate from $\mathcal{D}$ that does not belong to C'. Without loss of generality, suppose $d_1 \notin C'$. Now consider the set of votes given by $V' := \{v(1, X) \mid X \in \mathcal{F}\}$. We claim that there is at least one voter $v_j \in V'$ such that C' does not contain any approved candidate for v_j. Indeed, if not, then given that $d_1 \notin C'$, it is straightforward to verify that U' is a subset candidates of size at most k which intersect every set in $\mathcal{F}$, contradicting our assumption that $\mathcal{F}$ has no hitting set of size at most k. To see this, observe that every vote in V' that has zero misrepresentation is necessarily represented by a candidate from U', since $d_1 \notin C'$, and therefore, the sets corresponding to all of these votes are hit by U'. ■

Recall that the committee $\mathcal{D}$ has a misrepresentation score of one. Hence, using Lemma 34, since $\mathcal{F}$ has no hitting set of size k, $\mathcal{D}$ is a winning committee, as desired.

In the reverse direction, we assume $\mathcal{D}$ is a winning committee for the constructed CCCW instance. Hence, the optimal misrepresentation score is one. It is easy to see that if there exists a hitting set of size at most k, then the misrepresentation score for a committee C' formed using candidates corresponding to the elements of the hitting set is zero (since in that case, the committee will contain at least one approved candidate for each voter). Thus, if there exists a hitting set, $\mathcal{D}$ is not a winning committee – this contradicts the case we are in. Therefore, $\langle U, \mathcal{F}; k\rangle$ was a YES-instance of c-3-HITTING SET.

This completes the argument of equivalence. ■

The MONROE WINNER VERIFICATION problem is coNP complete for ℓ_1-Borda misrepresentation score in the setting of approval ballots. The argument for membership is similar to the one described at the beginning of this section, and to show the hardness, we again reduce from c-BOEHS. Due to similarity of arguments with Theorem 20, we omit the details.

With the results from Sections 5.1, 5.2 and 5.3, we have proved the following theorem.

Theorem 21 WINNER VERIFICATION *for Chamberlin-Courant and Monroe is coNP-complete in the setting of approval ballots and rankings. In the latter setting, the result holds for the ℓ_1 and ℓ_∞-Borda misrepresentation functions.*

5.4 Efficient Algorithms on Restricted Preferences

In this section, we consider the WINNER VERIFICATION problem on the single-peaked (SP) and the single-crossing (SC) domain for the Chamberlin-Courant voting rule. We refer the reader to Section 5.0.2 for the definitions of SP, SC domains.

Recall that our definition of election winner allows more than one committee to be the simultaneous winners for the election. In particular, for scoring based rules such as the Chamberlin-Courant rule and the Monroe rule, all the committees with size at most k achieving the minimum dissatisfaction score belong to the set of winning committees. Recall that the dissatisfaction score for a committee is computed by assigning each voter to a candidate in the committee and then accounting for the position of this assigned candidate in the voter's preference order.

For a committee C, let $score(C)$ be the dissatisfaction score of C for the given election. For the Chamberlin-Courant rule, we can find a winning committee using dynamic programming algorithms for each of single-peaked [68] and single-crossing [69] domains. Given a committee, one can compute its dissatisfaction score in $O(mn)$ worst-case time for the Chamberlin-Courant rule by assigning each voter to its most preferred candidate in the committee. Let opt be the optimal dissatisfaction score, i.e., dissatisfaction score corresponding to a winning committee. Hence, given an election instance with single-peaked or the single-crossing preference domain, opt can be computed in polynomial time.

Given an instance of the Winner Verification problem with an input election $E = (C, V)$ with SP or SC preference domain, and a k-sized committee S; first we compute the optimal dissatisfaction score opt in polynomial time. Next, we compute $score(S)$. At this stage, it is easy to observe that S is a winning committee if and only if $score(S) = opt$. Results from [68] and [70] also give polynomial-time algorithms for computing a winning committee for Chamberlin-Courant voting rule on single-peaked and single-crossing preferences respectively for ℓ_1-Borda dissatisfaction score in the case of rankings. Hence, We summarize our result as follows:

Theorem 22 CHAMBERLIN-COURANT WINNER VERIFICATION *can be solved in polynomial time for both* ℓ_1, ℓ_∞*-Borda dissatisfaction score on each of single-peaked and single-crossing domains.*

Chapter 6

Candidate Winner Verification Problem

In this chapter, we consider the CANDIDATE WINNER problem. Recall that in the CANDIDATE WINNER problem, we are given an election $E = (C, V)$[1], a committee size k, and a candidate $c \in C$, we ask if c belongs to *some* optimal k-sized committee, in other words, if there exists $S \subseteq C$ such that $c \in S$, $|S| = k$, and S is a winning committee. We consider this problem under Chamberlin-Courant and Monroe voting rule under min-sum (ℓ_1) and min-max (ℓ_∞) aggregation function for rankings, and min-sum aggregation for approval ballots. We refer the reader to Section 5.0.2 for the formal definitions of voting rules and aggregation functions. For the Chamberlin-Courant rule, we refer to the above problem as CHAMBERLIN-COURANT CANDIDATE WINNER CCCW and for the Monroe rule we refer to it as MONROE CANDIDATE WINNER MCW.

Our main result in this chapter is the θ_2^P-hardness for the CANDIDATE WINNER problem in (almost) all the cases we consider for general preferences and we complement this hardness results by solving the problem in polynomial time when input preferences follow

[1] Recall that for Chapter 6 we represent the set of candidates by C.

the single-peaked or single-crossing property (refer to Section 5.0.2 for the formal definitions of the single-peaked/single-crossing preference domains). Election problems have a long history of computational hardness. For example, Bartholdi et al. [71] showed that determining winners for many, otherwise excellent, voting rules are NP-hard. Prominent examples of such single winner (k=1) rules include Kemeny's voting rule [72], and Lewis Caroll's rule (Dodgson Rule, [61]). Moreover, some of these single winner rules seem to be substantially harder than any NP-completeproblem — they are complete for the complexity class $P_{\|}^{NP}$ [67]. Papadimitriou and Zachos [64] were the first to introduce the class $P_{\|}^{NP}$. Any language in this class can be decided in polynomial time using a polynomial number of *parallel* access to an NPoracle. Notice that, *parallel* access forbids adaptive queries and only allows 'batch' queries to an NPoracle (see Section 5.0.2 for more details).

6.0.1 Results and Organization of the Chapter

Our main result is to show θ_2^P-completeness for the Candidate Winner problem. The argument for membership in θ_2^P is nontrivial in this case (as opposed to Winner verification membership in coNP from the previous chapter), so we defer the details to the individual sections. In Section 6.1 and 6.2 we consider the Chamberlin-Courant and Monroe rule, respectively and show θ_2^P-completeness in the case of min-sum (ℓ_1) and min-max (ℓ_∞) ranking. Next, we consider the case of approval elections in Section 6.3 and show that the problem still remains hard for both CC and Monroe even in this case. Finally, in Section 6.4, we show that the Candidate Winner can be solved in polynomial time in all scenarios and complement our hardness results.

We note that the Candidate Winner problem was shown to be Θ_2^P-complete for the Chamberlin-Courant rule, for the utilitarian aggregation mechanism, in the setting of rankings [59][2]. We show the analogous result for the egalitarian version of the rule, and also for both aggregation mechanisms in the context of the Monroe rule. Although these reductions are executed in a similar spirit, the different settings do require nontrivial techniques in the constructions.

6.1 Chamberlin-Courant Voting Rule

In this section, we consider the Chamberlin-Courant Candidate Winner problem. For a given voting rule, the input to the problem is $\langle C, V, c, k\rangle$, and the question is if there exists a winning committee of size k containing the candidate c under the given voting rule for the election $E = (C, V)$. We investigate two variants of CCCW – the ℓ_1-Borda, and ℓ_∞-Borda misrepresentation functions for rankings.

We first consider the case of the ℓ_1-Borda misrepresentation. We start with a membership argument, and later establish the hardness. Recall the class θ_2^P, the membership to the class can be shown by giving a polynomial-time algorithm with parallel access to NP oracle. We use the NP oracle queries to obtain answer to the following variant of CCCW: so we ask if there exists a k-sized committee containing c which has misrepresentation score at most s (for some integer s). Note that the worst possible misrepresentation score in an instance with m candidates and n voters is mn so we only need polynomial guesses to obtain the optimal score. Thus, our algorithm $\mathcal{P}$ "guesses" the target score and finds the score of the optimal CC committee, and then the algorithm also finds an optimal CC committee that contains c. Note that overall, we only need $2mn$ queries. Observe that with an input target misrepresentation score, both types of our queries above belong to NP. Once we find the global optimal score and the optimal score of when c belongs to the committee, we compare the two optimal scores to decide whether there exists an optimal

[2]This result was independently discovered by the authors of the present work, while following up on a weaker version of the from [60].

CC committee of size k containing c.

We now turn to the reduction to demonstrate hardness. We recall that the following result was independently shown in [60].

Theorem 23 CHAMBERLIN-COURANT CANDIDATE WINNER *is* Θ_2^P*-hard for the* ℓ_1*-Borda misrepresentation score.*

Proof: We reduce from the Θ_2^P-complete problem VERTEX COVER MEMBER. Recall that we are given a graph $G := (V, E)$, and a vertex $w \in V$, the question is if there exists a minimum sized vertex cover containing w. Given an instance $\langle G := (V, E), w\rangle$ of VERTEX COVER MEMBER we construct an instance of CC CANDIDATE WINNER as follows. Let the set of candidates be $C := C_v \cup D \cup D'$, where C_v denotes the set of n candidates corresponding to vertices of G, and D and D' denote *type I* and *type II* dummy candidates, respectively. Let Δ denote a set of $nm + n^2 + 1$ type I dummy candidates, and Δ' denote a set of $(n+2)m+2n^2+2$ type II dummy candidates. We note that the subsets of dummy candidates specified explicitly in different votes are always chosen so that there are no repeated dummy candidates in the explicitly defined blocks, in other words, the chosen dummy candidates are always distinct. Also, D (D') is the union of all the Δ's (Δ''s) specified in the profile, which is given by the following three blocks of voters:

- **Block 1**: We construct m votes corresponding the edges in G. For an edge (u, v) we add:

$$c_u \succ c_v \succ \Delta' \succ C_v \setminus \{c_u, c_v\} \succ \text{rest}$$

where "rest" denotes the set of remaining candidates placed in an arbitrary order.

- **Block 2:** For the desired vertex w from the VERTEX COVER MEMBER instance, we pick an arbitrary edge incident on w (say (w, x)) in G, and add $m+1$ copies of the following vote:

$$c_w \succ c_x \succ \Delta \succ C_v \setminus \{c_w, c_x\} \succ \text{rest}$$

- **Block 3:** We add n votes of the form:

$$\Delta \succ c_w \succ C_v \setminus c_w \succ \text{rest}$$

In the constructed CC Candidate Winner instance, we ask if there exists an optimal committee of size n containing c_w. Before showing the equivalence of the two instances, we establish the following lemma.

Lemma 35 *Let q be the size of an optimal vertex cover in G. Then, following holds for any committee C' of size n in the constructed election instance:*

1. *If C' contains $d' \in D'$, then C' is not an* optimal *committee.*
2. *If C' does not contain exactly q candidates corresponding to an optimal vertex cover, then C' is not optimal.*

Proof: Suppose the candidate d' in C' belongs to the Δ' that appears after c_a, c_b for a voter in *Block 1* corresponding to the edge (a, b) in G. Consider a committee $\mathcal{C}$ formed by replacing d' with c_a in C'. Since c_a appears before d' in all votes, misrepresentation($\mathcal{C}$) $<$ misrepresentation(C') (note that for any optimal committee, every candidate in the committee represents at least one vote). Hence, no optimal committee contains any candidate from D'. This completes the proof of the first statement.

Towards showing the second statement, we first analyze the CC score of a committee $\mathcal{C}$ which contains q candidates corresponding to an optimal vertex cover, and the remaining $(n - q)$ candidates from D which appear in the top positions of $(n - q)$ votes in *Block 3*. The misrepresentation for $\mathcal{C}$ in Block 1 is at most m, in Block 2 it is at most $m + 1$.

In Block 3, for those votes that are not represented by the top candidate already, the misrepresentation for $\mathcal{C}$ is at most $(|\Delta| + n - 1)$ per vote since all the voters in Block 3 are represented by some candidate in C_v in the worst case. Hence, we have that misrepresentation$(\mathcal{C}) \leq 2m + 1 + (|\Delta| + n - 1) \times q$.

Now, let C' be an optimal committee which does not contain q candidates corresponding to some optimal vertex cover. Since C' is optimal, $C' \cap D' = \phi$. We will consider following cases:

- $|C' \cap C_v| > q$**:** In this case, C' contains at most $(n - q - 1)$ candidates from D. Hence, the misrepresentation of C' from Block 3 is at least $|\Delta| \times (q + 1)$ which is greater than the misrepresentation score for $\mathcal{C}$. This contradicts the optimality of C'.

- $|C' \cap C_v| \leq q$**:** Since the size of an optimal vertex cover is q, any committee C' with at most q candidates from C_v does not include any candidates corresponding to the endpoint of at least one edge due to the case we are in (i.e. C' does not contain candidates corresponding to optimal vertex cover). Hence, C' incurs a misrepresentation of at least $|\Delta'|$ from one of the votes in Block 1 which implies misrepresentation(C') > misrepresentation$(\mathcal{C})$.

This completes the proof for Lemma 35. ■

We now turn to the proof of equivalence. In the forward direction, given an optimal vertex cover of size q containing w, we construct an optimal committee C' by choosing q candidates corresponding to the vertex cover, and $(n - q)$ candidates from the set D which appears in the top position of exactly $(n-q)$ votes from Block 3. By Lemma 35, we already know that any optimal committee must contain candidates corresponding to an optimal sized vertex cover. Therefore, it suffices to show that committees corresponding to optimal vertex covers not containing c_w are not optimal. Indeed, this follows from the fact that in Block 2, c_w is the top candidate in exactly $(m+1)$ votes, and in Block 3, c_w leads all other candidates from the set C_v. Hence, it is easy to verify that an optimal committee must contain c_w.

In the reverse direction, given an optimal committee C' containing c_w, we need to construct an optimal vertex cover for G which includes the vertex w. Since C' is optimal, using Lemma 35 we know $C' \cap C_v$ is an optimal vertex cover of G. Also we are given that $c_w \in C'$; therefore, we have that the vertex cover corresponding to C' is an optimal vertex cover containing c_w, as desired. ■

Now, we show that CCCW is Θ_2^P-hard in the setting of rankings for the ℓ_∞-Borda misrepresentation function. The argument for membership is similar to the previous case with only difference that in this case the worst case score is m.

Theorem 24 Chamberlin-Courant Candidate Winner *is* Θ_2^P*-hard for the* ℓ_∞*-Borda misrepresentation score.*

Proof: As before, we reduce from Vertex Cover Member. Given an instance $\langle G := (V, E), w\rangle$ of Vertex Cover Member we construct an instance of CCCW as follows. Let the set of candidates be $C := C_v \cup D \cup D'$, where C_v denotes the set of n candidates corresponding to vertices of G, and D and D' denote *type I* and *type II* dummy candidates respectively. Let Δ denote a set of $m+n+1$ type I dummy candidates, and Δ' denote a set of 2 type II dummy candidates. We note that the subsets of dummy candidates specified explicitly in different votes are always chosen so that there are no repeated dummy candidates in the explicitly defined blocks, in other words, the chosen dummy candidates are always distinct. We construct the set of voters as the following three blocks:

- **Block 1**: We construct $(n+2)$ copies of each of m votes corresponding to the edges in G. Specifically, for an edge (u, v) we add $(n+2)$ copies of the vote:

$$c_u \succ c_v \succ \Delta \succ C_v \setminus \{c_u, c_v\} \succ \text{rest},$$

where *rest* denotes the set of remaining candidates in some arbitrary order.

- **Block 2:** We add the following n votes:

$$\begin{aligned}
&v_1 := \Delta' \succ d_{1,1} \succ d' \succ \Delta \succ C_v \succ \text{rest}\\
&\vdots\\
&v_i := \Delta' \succ d_{i,1} \succ \ldots \succ d_{i,i} \succ d' \succ \Delta \succ C_v \succ \text{rest}\\
&\vdots\\
&v_n := \Delta' \succ d_{n,1} \succ \ldots \succ d_{n,n} \succ d' \succ \Delta \succ C_v \succ \text{rest}
\end{aligned}$$

- **Block 3:** We also add $(n+2)$ copies of the following vote to force d' in any optimal committee:

$$d' \succ \Delta \succ \text{rest}$$

In the constructed CCCW instance, we ask if there exists an optimal committee of size $n+1$ containing c_w. This completes the construction for our reduction. We now state a lemma analogous to Lemma 35.

Lemma 36 *Let q be the size of an optimal vertex cover in G such that $q \geq 2$. Then, the following holds for any committee C' of size $n+1$ in the constructed election instance:*

1. *Any optimal committee contains candidate d'*
2. *If C' contains $d \in D$, then C' is not an* optimal *committee*

3. *If C' does not contain exactly q candidates corresponding to an optimal vertex cover, then C' is not optimal.*

Proof: Consider a committee $\mathcal{C}$ which contains q candidates corresponding to an optimal vertex cover, candidate d', and $n-q$ candidates $d_{j,q+1}$ for $j \in \{n-q+1, n-q+2, \ldots, n\}$. It is easy to see that the misrepresentation for this committee is $q+2$ (note that misrepresentation from Block 1 is at most one since we pick candidates corresponding to vertex cover, and misrepresentation from Block 2 is exactly $q+2$ since votes v_j for $j \in \{n-q+1, n-q+2, \ldots, n\}$ are represented by candidate in position $q+3$) and the votes v_j for $j \in \{1, 2, \ldots, n-q\}$ are represented by d'.

Given committee $\mathcal{C}$, from the construction of Block 3, it is clear that d' belongs to any *optimal* committee. Note that in an optimal committee, every candidate represents at least one voter. Hence, given committee $\mathcal{C}$ and committee size $n+1$, if a candidate $d \in D$ belongs to C' and represents some vote in Block 1 then, we know that C' does not contain candidates corresponding u, v such that $c_u, c_v \in C_v$. Given we have $n+2$ copies of each edge vote, C' is not an optimal committee.

It is easy to see that if C' does not contain candidates corresponding to vertex cover then misrepresentation score of C' is at least $n+m+1$ and C' is not an optimal committee (since we have $n+2$ copies of every edge). Next, if C' contains strictly more than q vertex candidates then the misrepresentation score of C' from Block 2 is strictly greater than $q+2$ which implies C' is not optimal. This completes the proof of Lemma 36. ∎

Next, we show proof of equivalence. In the forward direction, given an optimal vertex cover V' of size q containing w, we construct a committee $\mathcal{C}$ as described in the paragraph one of the proof of Lemma 36. We claim that $\mathcal{C}$ is an optimal committee. Using Lemma 36 we know that any optimal committee contains exactly q candidates corresponding to an optimal vertex cover, and contains candidate d'. From the structure of Block 2, it is easy to observe that in order to minimize the overall misrepresentation from Block 2, the only way is to choose the candidates from position $q+3$ for voters v_j for $j \in \{n-q+1, n-q+2, \ldots, n\}$. Note that first q voters from Block 2 are represented by d' while staying within the misrepresentation limit of $q+2$.

In the reverse direction, let C' be an optimal committee containing c_w. Using Lemma 36, we know that $C' \cap C_v$ is an optimal vertex cover. Hence, we can recover an optimal vertex cover containing c_w, concluding the argument.

■

6.2 Monroe Voting Rule

In this section, we turn to the MONROE CANDIDATE WINNER problem. Recall that the input is $\langle C, V, c, k \rangle$, and the question is if there exists a Monroe winning committee of size k containing c. We demonstrate that the problem is complete for Θ_2^P in the setting of rankings for both the ℓ_1-Borda and ℓ_∞-Borda misrepresentation functions.

Since the argument of membership is similar as presented at the start of Section 6.1, we omit that for brevity. We first consider the case of the ℓ_1-Borda misrepresentation and demonstrate the hardness.

Theorem 25 MONROE CANDIDATE WINNER *is Θ_2^P-hard in the case of rankings for the ℓ_1-Borda misrepresentation score.*

Proof: We again reduce from the Θ_2^P-complete problem VERTEX COVER MEMBER. Given an instance $\langle G := (V, E), w \rangle$ of VERTEX COVER MEMBER we construct an instance of MONROE CANDIDATE WINNER as follows. Let the set of candidates be $C := C_v \cup D \cup D' \cup S$, where C_v denotes the set of n candidates corresponding to vertices of G and D and D' denote *type I* and *type II* dummy candidates respectively. Let Δ

denote a set of n^4m type I dummy candidates, and Δ' denote a set of $2(n^4m)$ type II dummy candidates. We note that the subsets of dummy candidates specified explicitly in different votes are always chosen so that there are no repeated dummy candidates in the explicitly defined blocks, in other words, the chosen dummy candidates are always distinct. Also, D (D') is the union of all the Δ's (Δ''s) respectively specified in the profile, which is given by the following five blocks of voters:

- **Block 1**: We construct m votes corresponding to the edges in G. For an edge (u, v) we add:

$$c_u \succ c_v \succ \Delta' \succ C_v \setminus \{c_u, c_v\} \succ \text{rest}$$

  where "rest" denotes the set of remaining candidates placed in an arbitrary order.

- **Block 2:** For the desired vertex w from the Vertex Cover Member instance, we pick an arbitrary edge incident on w (say (w, x)) in G, and add $m+1$ copies of the following vote:

$$c_w \succ c_x \succ \Delta' \succ C_v \setminus \{c_w, c_x\} \succ \text{rest}$$

- **Block 3:** We add n votes of the form:

$$\Delta \succ c_w \succ C_v \setminus \{c_w\} \succ \text{rest}$$

- **Block 4:** For each candidate $d_j \in D_s$, we add the vote:

$$d_j \succ \Delta' \succ \text{rest}.$$

- **Block 5:** Let D_α be a subset of dummy candidates $d \in D$ such that d appears in the top position for one of the votes in Block 3. Note that $|D_\alpha| = n$. Furthermore, let $\mathcal{N} = 2(m+n+1)$. For each $v \in \{C_v \cup D''\}$ and $\ell \in [\mathcal{N}]$, we add the following vote:

$$v \succ D_s \succ \Delta' \succ \text{rest}$$

In the constructed Monroe Candidate Winner instance, we ask if there exists an optimal committee of size $2n+1$ containing c_w. Before showing the equivalence of the two instances, we establish the following lemma.

Lemma 37 *Let q be the size of an optimal vertex cover in G. Then, the following holds for any optimal committee C' of size $2n+1$ in the constructed election instance:*

1. *C' does not contain any $d' \in D'$.*
2. *$S \subset C'$.*
3. *C' contains exactly q candidates corresponding to an optimal vertex cover.*

Proof: First, we analyze the Monroe score of a committee $\mathcal{C}$ which contains all $n+1$ special candidates S, q candidates corresponding to an optimal vertex cover (S'), and the remaining $(n-q)$ candidates from D which appear in the top positions of $(n-q)$ votes in *Block 3*. Note that in any Monroe committee C', each candidate represents exactly $\mathcal{N}$ votes. We now describe the Monroe assignment for $\mathcal{C}$. Each vote in Block 1 and 2 is represented by one of the top two candidates such that the corresponding vertex (v) belongs to the vertex cover (S'). The misrepresentation for $\mathcal{C}$ in Block 1 is at most m, and in Block 2 it is at most $m+1$. In Block 3, exactly $n-q$ votes are represented by their first choice. For those votes that are not represented by the top candidate already, the misrepresentation for $\mathcal{C}$ is at most $(|\Delta|+n-1)$ per vote since all the votes in Block 3 are represented by the candidate in C_v for that vote in the worst case. In Block 4, all votes are represented by their top choice yielding zero misrepresentation. Votes in Block

5 are represented as follows: For each candidate c_i corresponding to a vertex $u \in S'$, if c_u represents t votes from first 3 blocks, then c_u also represents $(\mathcal{N} - t)$ votes from Block 5 among the ones she appears at the first position. Similarly, for $d \in \{D \cap \mathcal{C}\}$, d represents $\mathcal{N} - 1$ votes among the ones she appears at the top position. Next, each special candidate $s \in S$ represent $(\mathcal{N} - 1)$ votes in Block 5, yielding misrepresentation score at most $(n + 1)(\mathcal{N} - 1)$ for each s. Hence, the total misrepresentation for $\mathcal{C}$ is strictly less than $n^4m + 3mn^2 + 3n^3 < 2(n^3m^2 + n^4m)$ for large enough n.

Towards showing the first statement, consider a committee C^* which contains $d' \in D'$. In any Monroe assignment, d' has to represent $\mathcal{N}$ votes. Observe that d' appears in first (n^4m) positions exactly once, hence, misrepresentation$(C^*) >$ misrepresentation$(\mathcal{C})$. To show the second statement, consider a committee C^* which excludes a special candidate $s \in S$. It is easy to see that misrepresentation$(C^*) >$ misrepresentation$(\mathcal{C})$ even if we only consider misrepresentation from a single vote from Block 4 with s at the first position.

We now turn to statement 3. Now, let C^* be an optimal committee which does not contain q candidates corresponding to some optimal vertex cover. We use statements 1 and 2 to analyze the following two cases:

- **$|C^* \cap C_v| > q$:** In this case, C^* contains at most $(n - q - 1)$ candidates from D. Hence, the misrepresentation of C^* from Block 3 is at least $|\Delta| \times (q + 1)$ which is greater than the misrepresentation score for $\mathcal{C}$. This contradicts the optimality of C^*.

- **$|C^* \cap C_v| \leq q$:** Since the size of an optimal vertex cover is q, any committee C^* with at most q candidates from C_v does not include any candidates corresponding to the endpoint of at least one edge due to the case we are in (i.e., C^* does not contain candidates corresponding to an optimal vertex cover). Hence, C^* incurs a misrepresentation of at least $|\Delta'|$ from one of the votes in Block 1 which implies

$$\text{misrepresentation}(C^*) > \text{misrepresentation}(\mathcal{C}).$$

This completes the proof for Lemma 37. ■

We now turn to the proof of equivalence. In the forward direction, given an optimal vertex cover of size q containing w, we construct an optimal committee C' by choosing q candidates corresponding to the vertex cover, $(n-q)$ candidates from the set D which appears in the top position of exactly $(n-q)$ votes from Block 3, all $n+1$ special candidates S. We compute the Monroe assignment of C' is the same way we did for committee $\mathcal{C}'$ in Lemma 37. By Lemma 37, we already know that any optimal committee must contain all candidates from S, and candidates corresponding to an optimal sized vertex cover. Therefore, it suffices to show that committees corresponding to an optimal vertex covers not containing c_w are not optimal. Indeed, this follows from the fact that in Block 2, c_w is the top candidate in exactly $(m+1)$ votes, and in Block 3, c_w leads all other candidates from the set C_v. Hence, it is easy to verify that an optimal committee must contain c_w.

In the reverse direction, given an optimal committee C' containing c_w, we need to construct an optimal vertex cover for G which includes the vertex w. Since C' is optimal, using Lemma 37 we know $C' \cap C_v$ is an optimal vertex cover of G. Since we are given that $c_w \in C'$, we have that the vertex cover corresponding to C' is an optimal vertex cover containing c_w, as desired. ■

Next, for consider ℓ_∞-Borda misrepresentation function, the membership argument to θ_2^P is similar to previous case. We note that our reduction is similar as well to reduction in Theorem 24 so we skip the details.

6.3 Elections with Approval Preferences

Next, we turn to the case of approval ballots. We note that the argument of memberships is analogues to the case of ranking preferences so we skip the details. We now show the θ_2^P-harndess.

Theorem 26 CHAMBERLIN-COURANT CANDIDATE WINNER *is* Θ_2^P*-hard even in the case of approval ballots for the* ℓ_1*-Borda misrepresentation score.*

Proof: We reduce from VERTEX COVER MEMBER. We note that the argument in this case is relatively simpler. Given an instance $\langle G := (V, E), w \rangle$ of VERTEX COVER MEMBER we construct an instance of CCCW as follows. Let the set of candidates be, $C := C_v \cup D$ where C_v denotes the set of n candidates corresponding to vertices of G, and D denotes the set of n dummy candidates.

- **Block 1**: we construct $2n$ copies of each of the m votes corresponding to the edges in G. In particular, for $i \in [m]$, let (u, v) be the i^{th} edge. Then, we add $2n$ copies of vote v_i such that v_i approves the candidates corresponding to the edge (c_u, c_v), and she disapproves all other candidates.

- **Block 2:** For the desired vertex w from VERTEX COVER MEMBER instance, we add the vote v_w such that v_w disapproves all candidates except c_w.

- **Block 3:** We add two copies of each of the following n votes $\{v_1, v_2, \ldots, v_n\}$ such that v_i only approves a unique dummy candidate d_i and disapproves all other candidates.

In the constructed CCCW instance, we ask if there exists an optimal committee of size n containing c_w. Next, we establish the following structural lemma.

Lemma 38 *Let q be the size of an optimal vertex cover in G. Then, for any optimal committee C' of size n, C' contains exactly q candidates corresponding to an optimal vertex cover.*

Proof: We give a proof by contradiction. Consider a committee $\mathcal{C}$ which contains q candidates corresponding to an optimal vertex cover, and $(n-q)$ candidates from D. The misrepresentation for $\mathcal{C}$ in block 1 is zero, and in block 2 it is at most 1. In block 3, the misrepresentation is exactly $2q$ since for exactly $2(n-q)$ votes in block 3, C' contains the dummy candidates approved by these voters. Hence, the overall misrepresentation$(\mathcal{C}) \leq 1 + 2q$.

Let C' be an optimal committee which does not contain q candidates corresponding to some optimal vertex cover. We will consider following cases:

- $|C' \cap C_v| > q$**:** In this case, C' contains at most $(n-q-1)$ candidates from D. Hence, the misrepresentation of C' from block 3 is at least $2 \times (q+1)$ which is greater than the misrepresentation score for $\mathcal{C}$. This contradicts the optimality of C'.

- $|C' \cap C_v| \leq q$**:** Since the size of an optimal vertex cover is q, any committee C' with at most q candidates from C_v does not include any approved candidates for n votes corresponding to at least one edge due to the case we are in (i.e. C' does not contains candidates corresponding to an optimal vertex cover). Hence, C' incurs a misrepresentation of at least $2n$ from $2n$ votes corresponding to an edge in Block 1 which implies misrepresentation$(C') >$ misrepresentation$(\mathcal{C})$. This completes the proof of Lemma 38.

■

Next, we turn to the proof of equivalence. In the forward direction, given an optimal vertex cover of size q containing w, we construct an optimal committee C' by choosing q

candidates corresponding to the vertex cover, and arbitrarily chosen $(n-q)$ candidates from the set D. By Lemma 38, we already know that any optimal committee must contain candidates corresponding to an optimal sized vertex cover (hence, at most $n-q$ dummy candidates). Therefore, it suffices to show that committees corresponding to an optimal vertex cover not containing c_w are not optimal. This is easy to see from a vote in Block 2 which only approves c_w. Hence, optimal committee must contain c_w.

In the reverse direction, given an optimal committee containing c_w, in the light of Lemma 38 we know $C' \cap C_v$ is an optimal vertex cover of G. Since we are given $c_w \in C'$, we have that the vertex cover corresponding to C' is an optimal vertex cover containing c_w, as desired. ∎

The Monroe Candidate Winner problem in θ_2^P-complete for ℓ_1-Borda misrepresentation score in the setting of approval ballots. The argument of membership is similar as before, and to show the hardness, we again reduce from the Vertex Cover Member problem. Due to similarity of arguments with Theorem 26, we omit the details.

We summarize our results from Sections 6.1, 6.2 and 6.3 in the following theorem:

Theorem 27 Candidate Winner *for Chamberlin-Courant and Monroe is complete for* Θ_2^P *in the setting of approval ballots and rankings. In the latter setting, the result holds for the* ℓ_1 *and* ℓ_∞*-Borda misrepresentation functions.*

6.4 Efficient Algorithms on Restricted Preference

In this section, we consider the Candidate Winner problem on the single-peaked (SP) and the single-crossing (SC) domain for the Chamberlin-Courant voting rule. We refer the reader to Section 5.0.2 for the definitions of SP, SC domains.

We start with ℓ_1-Borda dissatisfaction score for the single-peaked preference domain.

Let $E = (C,\ V),\ k,\ c$ be the input to a CCCW instance where the election contains single-peaked preferences of the voters, k is the size of the committee, and c is the desired input candidate. We compute a winning committee, and the optimal dissatisfaction score *opt* for E in polynomial time. Let

$$v' := c \succ (\{C \setminus c\} \text{ in any single-peaked order}).$$

We construct an election E' with same set of candidates, and with the following set of voters: $V' = V \cup (mn+1) \cdot v'$ i.e., V' contains V, and additionally contains $(mn+1)$ copies of the vote v'. Clearly, election E' contains single-peaked preferences with the same linear ordering of the candidates as in election E. Next, we compute the optimal dissatisfaction score opt' for election E'. We return YES for the CCCW instance if $opt = opt'$, otherwise we return NO.

Next, we show the correctness of our algorithm.

Lemma 39 *For a given instance $\langle C, V, k, c\rangle$ of* CCCW*, there exists a winning committee containing c iff $opt = opt'$.*

Proof: First, we assume that $opt = opt'$. We observe that $opt < mn+1$ as any committee can have a dissatisfaction score at most mn in E. Furthermore, for election E', any committee S, with $score(S) < mn+1$ has to include candidate c. This is because for any k-sized committee which does not contain c, $(mn+1)$ copies of vote v' contributes positively towards its dissatisfaction score. Since $opt = opt'$, the optimal committee for election E' (say S') includes candidate c. Hence, S' is a winning committee for election E containing candidate c.

In the other direction, we assume a k-sized winning committee S for the election E' containing candidate c. As $V \subset V'$, $score(S) \geq opt$. It is easy to see that S attains a dissatisfaction score opt in election E' since the additional $(mn+1)$ copies of v' do not incur any dissatisfaction as these voters are represented by their top choice c. This completes the proof of Lemma 39. ■

For ℓ_∞-Borda dissatisfaction score, let $E = (C, V)$, k, c be an input instance of CCCW where E contains single-peaked preferences. We compute a winning committee, and the optimal dissatisfaction score *opt* for E in polynomial time. Let

$$V^* := \{v \in V \mid v \text{ ranks candidate } c \text{ in top } opt \text{ positions}\}$$

i.e., $V^* \subseteq V$ is a set of voters which rank c in top *opt* positions. We construct an election E' with the same set of candidates, and set $V' = V \setminus V^*$. It is easy to see that election E' contains single-peaked preferences. Next, we compute the optimal dissatisfaction score opt' for the election E' with committee size $k-1$. Notice that if there exists an optimal committee S of size k containing c for the election E, then each voter in set $V \setminus V^*$ must have at least one candidate from a set of $k-1$ candidates $S \setminus c$ in top *opt* positions. Hence, we return Yes for the CCCW instance if $opt' \leq opt$, otherwise we return No. This completes the description, and a brief argument of correctness.

Next, we consider ℓ_1-Borda dissatisfaction score for the single-crossing preference domain.

Description of the algorithm. Let $\langle C, V, k, c\rangle$ be an instance of CCCW where $V = \{v_1, v_2, \ldots, v_n\}$ is a set of voters in a single-crossing order. For $0 < i \leq j \leq n$, $L_{i,j} = \{v_i, v_{i+1}, \ldots, v_j\}$ be a subset of V.

First, we compute the optimal dissatisfaction score *opt* for the input election instance. Next, we iterate through all $n(n+1)/2$ valid choices of i, j, and select the contiguous block $L_{i,j} \subseteq V$. For each iteration, we assume that candidate c represents the selected set of voters ($L_{i,j}$). Let $V_1 = \{v_{i'} \mid 1 \leq i' < i$, and $V_2 = \{v_{j'} \mid j < j' \leq n\}$. Furthermore, let $E_1 = (C, V_1)$, and $E_2 = (C, V_2)$ be two sub-elections. It is easy to observe that E_1, E_2

contains single-crossing preferences with the same order on voters. At this stage, we go over all choices of $0 \leq k_1, k_2 \leq k$ such that $k_1 + k_2 = k - 1$. We compute optimal committees S_1, S_2 of size k_1, k_2 for elections E_1, E_2 respectively. Let $\mathit{score}(c)$ be the total dissatisfaction score from block $L_{i,j}$ for when it is represented by candidate c. If for some "guess" of tuple (i, j, k_1, k_2), $\mathit{score}(S_1) + \mathit{score}(S_2) + \mathit{score}(c) = \mathit{opt}$ then we return YES, otherwise we return NO [3].

Refer to Algorithm 3 for the pseudocode of the algorithm. It is easy to see that our algorithm runs in polynomial time.

Algorithm 3 CCCW on single-crossing preferences

Input: $E = (C, V),\ k,\ c$
Compute optimal dissatisfaction score opt for SC election E
for $i \in \{1, 2, \ldots, n\};\ \ j \in \{1, 2, \ldots, n\}\ \mid\ 0 < i \leq j \leq n$ **do**
  Select contiguous block of votes $L_{i,j} \leftarrow \{v_i, v_{i+1}, \ldots, v_j\}$
  Let $V_1 \leftarrow \{v_1, \ldots, v_{i-1}\},\ V_2 \leftarrow \{v_{j+1}, \ldots, v_n\};\ \ E_1 = (C,\ V_1),\ \ E_2 = (C,\ V_2)$
  for $0 \leq k_1, k_2 \leq k$ such that $k_1 + k_2 = k - 1$ **do**
    Compute $S_1 \leftarrow$ optimal committee for E_1, $S_2 \leftarrow$ optimal committee for E_2
    Compute $\mathit{score}(c) \leftarrow$ total dissatisfaction score from $L_{i,j}$ represented by c
    if $\mathit{score}(S_1) + \mathit{score}(S_2) + \mathit{score}(c) = \mathit{opt}$ **then**
      return YES
return NO

Proof of correctness. We show the following lemma:

Lemma 40 *Algorithm 3 returns* YES *if and only if the input* CCCW *instance admits a winning committee containing c.*

Proof: First, we assume that S' is a k-sized winning committee containing c. Define $L_{i,j} \subseteq V$ as the contiguous block of voters represented by c where v_i, v_j are first and last voters in the block respectively. Note that $L_{i,j}$ always exists as Skowron et al. [69, Lemma5] showed that for single-crossing elections, there exists an assignment

[3]Note that Algorithm 3 assumes that for an optimal committee containing c, c represents at least one voter. We can eliminate the case when c does not represent any candidate by checking if the optimal scores for k and $k-1$-sized committees are the same

of voters such that each candidate in a winning committee represents a contiguous set according to the single-crossing ordering of voters. Let $C_1 \subseteq S'$ be the set of candidates representing voters $\{v_1, \ldots, v_{i-1}\}$, and $C_2 \subseteq S'$ be the set of candidates representing votes $\{v_{j+1}, \ldots, v_n\}$. For $k_1 = |C_1|$ and $k_2 = |C_2|$, let S_1 and S_2 be the optimal committees computed by Algorithm 3 for elections E_1 and E_2 defined by block $L_{i,j}$ (where i, j are defined by the committee S'). It is easy to see, $\mathit{score}(S_1) \leq \mathit{score}(C_1)$ for voters in election E_1. This is because S_1 is an optimal k_1-sized committee for E_1. Similarly, $\mathit{score}(S_2) \leq \mathit{score}(C_2)$. Note that $\mathit{score}(c)$ for the block of voters $L_{i,j}$ is the same in both the committee S' and the committee $S = S_1 \cup S_2 \cup c$ computed by Algorithm 3. Hence, $\mathit{score}(S_1) + \mathit{score}(S_2) + \mathit{score}(c) \leq \mathit{score}(C_1) + \mathit{score}(C_2) + \mathit{score}(c) = \mathit{score}(S') = \mathit{opt}$. Therefore, Algorithm 3 returns Yes.

In the other direction, we assume that Algorithm 3 returns Yes. In this case, we claim that $S = S_1 \cup S_2 \cup c$ is a valid k-sized winning committee. Assuming we are in the case when optimal dissatisfaction score for a $(k-1)$-sized optimal committee is strictly greater than a k-sized optimal committee, we observe that $S_1 \cap S_2 = S_1 \cap \{c\} = S_2 \cap \{c\} = \emptyset$. Otherwise, we can construct a k-sized committee with a dissatisfaction score strictly less than opt, a contradiction. Since $\mathit{score}(S_1) + \mathit{score}(S_2) + \mathit{score}(c) = \mathit{score}(S) = \mathit{opt}$, we conclude that S is a winning committee containing c. ■

Lemma 40 completes the argument for proof of correctness of Algorithm 3.

Since the dynamic programming algorithm to compute a winning committee in the case of ℓ_1-Borda dissatisfaction score on single-crossing domains for the Chamberlin-Courant voting rule [69] also works for ℓ_∞-Borda dissatisfaction score, Algorithm 3 works as it is for ℓ_∞-Borda misrepresentation score as well. We summarize our results as follows:

Theorem 28 Chamberlin-Courant Candidate Winner *can be solved in polynomial time for both ℓ_1, ℓ_∞-Borda dissatisfaction score on each of single-peaked and single-*

crossing domains.

Remark 2 *Note that most of our algorithms from Section 6.4 build on an efficient subroutine for computing a winning committee and, in turn, computing the optimal dissatisfaction score opt. Finding the optimal dissatisfaction score in the case of ℓ_1-Borda dissatisfaction function is known to be* NP-complete *for Monroe voting rule on both single-peaked [68], and single-crossing [69] preference domains. Skowron et al. [69] showed an efficient algorithm for computing a winning committee in case of ℓ_∞-Borda dissatisfaction score for Monroe voting rule on a restricted preference domain where the preferences are both single-peaked and single-crossing. We notice that we can efficiently solve the Winner Verification problem whenever opt can be computed in polynomial time; hence, it can be solved efficiently for the case of single-peaked and single-crossing preferences. Furthermore, Skowron et al. [69, Lemma 17] show that for each optimal committee, there exists an assignment of voters to candidates such that each candidate represents a contiguous set of voters according to the single-crossing ordering. Observe that we exploit exactly this property in Algorithm 3; hence, Algorithm 3 can be used to efficiently solve the* Candidate Winner *problem for ℓ_∞-Borda dissatisfaction function for Monroe rule on single-peaked and single-crossing preference domain. We pose the question of computational complexity for the remaining scenarios (compared to the cases considered for the Chamberlin-Courant rule) in the case of Monroe voting rule as an open problem.*

We summarize our results from Section 5.4 and Section 6.4 in the following theorem:

Theorem 29 Winner Verification *and* Candidate Winner *for Chamberlin-Courant are polynomial-time solvable for each of single-peaked and single-crossing preference domains in the setting of rankings. The result holds for the ℓ_1 and ℓ_∞-Borda misrepresentation functions.*

Chapter 7

Fair Covering of Points in Euclidean Space

In this chapter, we study a problem of fair covering of heterogeneous points using unit radius balls in $\mathbb{R}^d$. As described earlier (in Section 1.1), given a set P of n points in $\mathbb{R}^d$ each of which is colored by one of t colors, the fair covering problem aims to cover the maximum number of points using k unit-radius balls such that the coverage for each color is in *proportion* to its size. More precisely, let $\mathcal{C}$ be a family of k unit radius balls, c_i be the number of the points of color i that are covered by $\mathcal{C}$, and n_i be the total number of points of color i, for $i \in \{1, \ldots, t\}$. Then we say that the covering $\mathcal{C}$ is *fair* if

$$\lfloor \rho_i \cdot c^* \rfloor \leq c_i \leq \lceil \rho_i \cdot c^* \rceil$$

for all $i \in \{1, \ldots, t\}$, where $c^* = \sum_{i=1}^{t} c_i$ and $\rho_i = n_i/n$ for $i \in \{1, \ldots, t\}$. Among all *fair* coverings, we want the one that maximizes the total coverage c^*. We note that fair coverings always exist, because an empty covering trivially satisfies the fairness condition but covers no points.

Achieving strict fair covering can be computationally hard, so we also define the notion of *approximately fair* covering. A covering $\mathcal{C}$ is called ε-fair for some $\varepsilon \in [0, 1]$, if

$$(1 - \varepsilon) \cdot \lfloor \rho_i \cdot c^* \rfloor \leq c_i \leq (1 + \varepsilon) \cdot \lceil \rho_i \cdot c^* \rceil$$

for all $i \in \{1, \ldots, t\}$. The goal of the approximately fair covering problem is then to find an ε-fair covering that maximizes the number of covered points.

The topic of algorithmic *fairness* has received significant attention recently [73, 74, 75, 76, 77, 78, 79, 80], especially with the increasing use of machine learning in policy and decision making. Our work in this chapter explores the computational implications of fairness as a constraint in geometric optimization by focusing on the specific problem of covering by unit balls, or equivalently, fixed-radius facility location. The different colors in our input represent different demographic groups and *proportionality* is one of the most basic forms of fairness, requiring that each group's share in the solution is proportional to its size. The proportional fairness can be easily extended to *weighted* sharing by assigning nonuniform weights to different points or color classes and measuring fairness on the overall covered weights. The fair covering problem can also be viewed as *fair clustering* under the k-center measure when each cluster is constrained to have unit radius.

In this chapter, we investigate the aforementioned (approximately) fair covering problem under the *discreteness* and *bounded-ply* constraints defined below. We require the balls used in the covering to be chosen from a given candidate set of unit-radius balls (discreteness), and any point in the plane to be covered by at most p chosen disks where p is a given constant (bounded-ply). Formally, the input of the problem consists of a set P of n points in $\mathbb{R}^d$ each of which is colored with one of the t-colors, a candidate set $\mathcal{B}$ of m unit-radius balls in $\mathbb{R}^d$, a number k that is the budget of balls to be used, and a number p which is the bound on the ply of the covering. Our goal is to find a (approximately) fair covering for P using at most k balls in $\mathcal{B}$ such that any point in the plane (not only points from P) is covered by at most p balls and we cover the maximum number of points in P.

7.0.1 Results and Organization of the Chapter

Our main results are the following:

- We show that there exists an exact algorithm solving the fair covering problem in $\mathbb{R}^1$ in $O(m^3+m^2n^t)$ time. Alternatively, the problem can also be solved in $O(nm^k)$ time (Section 7.1).
- We show that the fair covering problem in $\mathbb{R}^1$ is NP-hard if the number of colors is part of the input. We also show that the problem is W[1]-hard parameterized by the number of covering balls k (Section 7.2).
- For a fixed $d \geq 2$ and a fixed number of colors, we present a PTAS (Polynomial Time Approximation Scheme) for the approximately fair covering problem (Section 7.3).
- We present an exact algorithm solving the *unconstrained (without discreteness or bounded ply)* fair covering problem in $\mathbb{R}^1$ in $O(n^{t+2})$ time (Section 7.4).

For our algorithmic results in Section 7.1 and in Section 7.3, first, we present our approach for the case of $p = 1$, i.e., the disks used in the covering are required to be disjoint; near the end of each of these sections, we add a discussion about necessary changes to adapt our approach for $p > 1$ (i.e., the bounded ply case).

7.1 Polynomial time Algorithm in One-dimension

In the first section we consider the problem in one dimension. Let $P = \{p_1, \ldots, p_n\}$ be a set of n points on the real line each of which belongs to one of the t color classes, and let $\mathcal{B} = \{B_1, \ldots, B_m\}$ be the candidate set of unit intervals on the line. (Technically speaking, a unit-radius ball in one dimension would be an interval of length 2, but a *unit-length* interval seems more natural, so that we shall use unit intervals in the following discussion. Note that the problem with intervals of length 2 is equivalent to the problem with unit intervals by simply scaling the points and the intervals.) Our goal is to cover the maximum number of points using at most k intervals in $\mathcal{B}$ forming a bounded ply cover under the fair covering constraint. We show that an optimal covering can be computed in polynomial time when the number t of colors is fixed.

For simplicity, we describe our algorithm for $t = 2$ and use red/blue as the two colors for easier reference. The extension to an arbitrary number of colors is straightforward. We start with the case when ply is equal to one i.e., the problem of finding a cover consisting of disjoint intervals in $\mathcal{B}$.

Given integers r and b, we define an (r, b)*-covering* to be a subset of $\mathcal{B}$ consisting of disjoint intervals that covers exactly r red and b blue points. An optimal (r, b)-covering is an (r, b)-covering that uses the minimum number of intervals. We solve the fair covering problem by computing an optimal (r, b)-covering for all $r, b \in \{1, \ldots, n\}$. Without loss of generality, we assume that the unit intervals $B_1, \ldots, B_m$ are sorted in the left-to-right order. Let $r(B_i)$ and $b(B_i)$ be the number of the red and blue points covered by B_i, respectively. For each $i \in \{1, \ldots, m\}$, let $\pi_i < i$ be the largest integer such that $B_{\pi_i} \cap B_i = \emptyset$; we assume $\pi_1 = 0$. We make a left-to-right pass over the set of input points and the intervals on the real line, and compute π_i, $r(B_i)$, $b(B_i)$ for all $i \in \{1, \ldots, m\}$.

Define $F[i, r, b]$ as the size of an optimal (r, b)-covering using only intervals in $\{B_1, \ldots, B_i\}$.

For the pairs (r, b) such that no (r, b)-covering exists, we set $F[i, r, b] = \infty$. It is easy to see that F satisfies the following recurrence.

Claim 3

$$F[i, r, b] \;=\; \min \left\{ \begin{array}{l} F[i-1, r, b] \\ 1 + F[\pi_i, r - r(B_i), b - b(B_i)] \end{array} \right\}$$

The above recurrence immediately allows us to compute the table F using dynamic programming, which is shown in Algorithm 4. The base case for the dynamic program is $F[i, 0, 0] = 0$ for all $i \in \{1, \ldots, m\}$ and $F[0, r, b] = \infty$ for all $r, b \in \{1, \ldots, n\}$.

Algorithm 4 Computing the F-table

Input: a set P of points on the line and the set of intervals $\mathcal{B}$
1: Compute $\pi_i, r(B_i), b(B_i)$ for $i \in \{1, \ldots, m\}$
2: Initialize m $\times\, r \times b$ sized table with value ∞
3: **for** $i \in \{0, \ldots, m\}$; $r, b \in \{0, \ldots, n\}$ **do**
4: $\quad F[i, r, b] \leftarrow min\{F[i-1, r, b], 1 + F[\pi_i, r - r(B_i), b - b(B_i)]\}$
5: **return** F

Lemma 41 *Algorithm 4 can be implemented in worst-case time $\mathcal{O}((n+m)\log(n+m) + mn^2)$.*

Proof: Sorting P and $\mathcal{B}$ takes $O((n+m)\log(n+m))$ time. Computing $\pi_i, r(B_i), b(B_i)$ for all $i \in \{1, \ldots, m\}$ takes additional linear time. After that the F-table can be computed in $O(mn^2)$ time. ■

Once the F-table is computed, we can solve the fair covering problem by checking all entries in the table for which the (r, b)-covering is fair and has $F[m, r, b] \leq k$. Among all such valid pairs, we return the pair (r^*, b^*) with the maximum $r^* + b^*$. Clearly, $c^* = r^* + b^*$ is the optimum of the problem instance. We therefore have the following result.

Theorem 30 *The disjoint fair covering problem in $\mathbb{R}^1$ with $t = 2$ colors can be solved in $O((n+m)\log(n+m) + mn^2)$ time.*

The dynamic program easily extends to the case of $t > 2$ colors, by using a $(t+1)$-dimensional DP table.

Theorem 31 *The disjoint fair covering problem in $\mathbb{R}^1$ can be solved in $O((n+m)\log(n+m) + mn^t)$ time.*

Extension to bounded ply

We now present a generalization of the dynamic programming algorithm from the previous section to find coverings with a constant ply. In other words, we now want to solve the (r,b)-covering problem when any point on the line can be covered by at most p intervals. Recall that we are looking for coverings $\mathcal{C}$ with at most k-disks. Let $\mathcal{C}(P) \subseteq P$ be the subset of points covered under $\mathcal{C}$. We say $\mathcal{C}$ is minimal if for $c_i \in \mathcal{C}$, the covering $\mathcal{C}'_i = \mathcal{C} \setminus c_i$ covers only a subset of points covered by $\mathcal{C}$, i.e., $\mathcal{C}'_i(P) \subset \mathcal{C}$. Next, we state a structural lemma for minimal coverings in one dimension shown in [81, Lemma 3].

Lemma 42 *Any minimal covering in $\mathbb{R}^1$ has ply at most* 2.

We refer the reader to [81] for the proof of Lemma 42.

Let $B_1, \ldots, B_m$ be the sorted order of the intervals from left to right. For each $i \in \{1, \ldots, m\}$, let $\pi_i < i$ be the largest index such that $B_{\pi_i} \cap B_{\pi_i+1} = \emptyset$; let the minimum value of π_i be 0. We make a left-to-right pass over the intervals, and store the index i in a set S if $B_i \cap B_{i+1} = \emptyset$ according to their order of discovery. Using the monotonicity property of π_i (i.e. $\pi_i \geq \pi_{i-1}$), we can compute π_i for $i \in \{2, \ldots, m\}$ with one pass over the set of intervals and the set S.

We retain the definition of $F[i,r,b]$ from the previous result. It is easy to see that $F[i,r,b]$ satisfies the following the recurrence.

Claim 4

$$F[i,r,b] \;=\; \min \left\{ \begin{array}{l} F[i-1,r,b], \\ \\ \displaystyle\min_{\pi_i \le j \le i} \ell_{i,j} + F[j, r-r_{i,j}, b-b_{i,j}] \end{array} \right\}$$

where $\ell_{i,j}$ is the minimum number of unit intervals required to cover the interval $[x^-, x^+]$ where x^- is the left endpoint of B_{j+1} and x^+ is the right endpoint of B_i, and $r_{i,j}$ (resp., $b_{i,j}$) is the number of red (resp., blue) points contained in $[x^-, x^+]$.

The idea of the recurrence is as follows: First term $F[i-1,r,b]$ considers the case when B_i is not included in the cover. Next, we take a minimum over at most m terms where we consider the case when B_i is included in the cover, and we "guess" the optimal length of a contiguous covered interval ending with B_i. Hence, our approach is exhaustive. The base cases for our recurrence are $F[i,0,0] = 0$ for all $i \in 0,\ldots,m$, and $F[0,r,b] = \infty$ for $r,b \in \{1,\ldots,n\}$. Next, we present a dynamic programming algorithm to efficiently compute the F table.

Algorithm 5 Computing the F-table

Input: $P, \mathcal{B}$
1: Compute π_i for $i \in \{1,\ldots,m\}$
2: Initialize m $\times r \times b$ sized table with value ∞
3: **for** $i \in \{0,\ldots,m\}$; $r,b \in \{0,\ldots,n\}$ **do**
4: $\quad F[i,r,b] \leftarrow min\{F[i-1,r,b], \displaystyle\min_{\pi_i \le j \le i} \ell_{i,j} + F[j, r-r_{i,j}, b-b_{i,j}]\}$
5: **return** F

Lemma 43 *Algorithm 5 can be implemented in worst-case time $\mathcal{O}(n^2m^2 + m^3)$.*

Proof: Sorting P and $\mathcal{B}$ takes $\mathcal{O}((n+m)\log(n+m))$ time. Computing π_i takes $\mathcal{O}(m)$ time using the procedure described earlier. We compute $\ell_{i,j}$ for all $\binom{m}{2}$ pairs in advance. For a given pair (i,j) such that $j < i$, we can find $\ell_{i,j}$ in $\mathcal{O}(m)$ time with a following procedure: We include interval B_j in the covering. At this stage, B_j is the rightmost interval in the constructed partial covering. Next, in each iteration, we include

$B_{j'}$ in the covering, such that $j' \leq i$ is the largest index such that $B_{j'}$ has a non-zero intersection with the rightmost interval in the partial covering. We then update the rightmost interval to $B_{j'}$. We repeat the procedure until we include B_i in the covering. Note that both B_j, B_i will always be included in a minimum sized covering assuming no two intervals are identical. Overall, all $\ell_{i,j}$ values are computed in $\mathcal{O}(m^3)$ time.

After that each F-table entry can be computed in $\mathcal{O}(m)$ time as follows: We can look-up $F[i-1, r, b]$ in a constant time. Next, for the case when B_i is included in the covering, we need to compute at most m terms ($i - \pi_i$ to be exact). Each of these terms involves a constant time look up for the respective $\ell_{i,j}$, and a previously computed F-table entry.

There are mn^2 entries in the F-table; hence, the overall running time is bounded by $\mathcal{O}(n^2m^2 + m^3)$. ■

Once the F-table is computed, we can solve the fair covering problem in the same way as described previously for $p = 1$ case. We therefore obtain the following result.

Theorem 32 *The fair covering problem in $\mathbb{R}^1$ with $t = 2$ colors can be solved in $\mathcal{O}(n^2m^2 + m^3)$ time.*

The result can be easily generalized to $t > 2$ colors, by using a $t + 1$ dimensional DP table.

Theorem 33 *The fair covering problem in $\mathbb{R}^1$ can be solved in $\mathcal{O}(n^tm^2 + m^3)$ time.*

Remarks. Recall that the fair covering problem we investigate is defined with the discreteness and bounded ply constraints. In fact, the problem without each of these two constraints can also be solved using similar dynamic programming approaches. We defer the details to the end (Section 7.4) because our main focus is the problem with discreteness and bounded ply constraints.

7.2 NP-hardness and W[1]-hardness of Fair Covering

In this section, we show that the one-dimensional fair covering problem is NP-hard if the number of colors t is large even for a disjoint cover (i.e., the case when $p = 1$). We also show that the problem is $\mathsf{W}[1]$-hard parameterized by the number of intervals k.

Theorem 34 *The one-dimensional fair covering problem with $\Omega(n)$ colors is* NP-hard.

Proof: We reduce the well-known EXACT COVER problem (which is known to be NP-complete [82]) to our problem. Given a ground set $\mathcal{U}$, a family $\mathcal{F}$ of subsets of $\mathcal{U}$, and an integer ℓ, the EXACT COVER problem is to decide if there exists a $\mathcal{S} \subseteq \mathcal{F}$ of size ℓ that contains each element of $\mathcal{U}$ exactly once. The construction is described below.

Construction. Given an instance of EXACT COVER with $\mathcal{U} = \{u_1, u_2, \ldots, u_n\}$, $\mathcal{F} = \{S_1, S_2, \ldots, S_m\}$, and an integer ℓ, we construct a set of points $\mathcal{P}$, and a set of centers $\mathcal{M}$ as follows. The i^{th} element of $\mathcal{U}$ is associated with color i; thus, there are n color classes. We also introduce an additional color 0, which we call *special*. The set of points is organized in the following three groups.

1. *Basic Points:* For each set $S_i \in \mathcal{F}$, we introduce $|S_i|$ points, placed arbitrarily within the interval $[3i, 3i + 1)$. Each point has the color of its element. The intervals corresponding to S_i and S_j, $i \neq j$, are distance 2 apart, which ensures that any unit interval of $\mathcal{B}$ can cover points of at most one such group.

2. *Balancers:* We add extra points for each color i to ensure that all colors $i = 1, 2, \ldots, n$ end up with the same number of points. Specifically, let f^* be the maximum number of sets to which an element belongs, and let f_i be the number of sets containing the element u_i. We introduce $f^* - f_i$ points of color i in the interval $[3(m + i), 3(m + i) + 1)]$.

Figure 7.1: Constructed fair covering instance for an Exact Cover instance $\mathcal{U} = \{1,2,3\}$, $\mathcal{F} = \{(1,3),(2),(1,2)\}$, $\ell = 2$. We introduce red (1), green (2), and blue (3) colors corresponding to the elements in the universe, and we also introduce cyan as the special color. First five points are introduced in the basic points group. Since $f^* = 2$ (where f^* is a maximum number of sets to which an element of $\mathcal{U}$ belongs to), next, we introduce one blue point so that each color except for cyan has exactly two points. At last, we introduce 4 cyan points as enforcers (since $f^* = \ell = 2$).

3. *Enforcers:* Finally, we introduce ℓf^* points of color 0 (special color), at locations $3(m+n+1), 3(m+n+2), \ldots, 3(m+n+\ell f^*)$. These are needed in our construction to enforce the fair covering condition. Refer to Figure 7.1.

Finally, the set of centers $\mathcal{M}$ is defined as follows.

- For each $S_i \in \mathcal{F}$, we add a center at $3i + 1/2$, which allows all points of that group to be covered by one unit interval.
- Each enforcer point is also a center. We do not need centers for the balancers—their role is primarily to make all color classes have equal size.

Finally, we fix the number of covering intervals to be $k = 2\ell$.

We now argue that the Exact Covering instance is a yes instance if and only if our fair covering instance admits a k-covering with at least $n + \ell$ points.

For the forward direction of the proof, suppose $\mathcal{S} \subseteq \mathcal{F}$ is an exact cover of size ℓ, and $\mathcal{T} = \{i \mid S_i \in \mathcal{S}\}$ be the set of indices. Then we build a covering $\mathcal{C}$ as follows. We place first ℓ intervals centered at $3i + 1/2$ for $i \in \mathcal{T}$, and the remaining ℓ intervals are placed at $3(m+n+j)$ for $j = 1, 2, \ldots, \ell$ covering one special colored point each. Since $\mathcal{S}$ is an exact cover, $\mathcal{C}$ contains exactly $n + \ell$ points. The covering is also fair, since all the colors $i = 1, 2, \ldots, n$ have the same number of points f^*, and the special color 0 has ℓf^* points. In the covering, each of the color classes $i = 1, 2, \ldots, n$ has one covered point and the special color has ℓ points.

For the reverse direction, let $\mathcal{C}$ be the fair covering with at least $n + \ell$ points. We observe that a fair covering necessarily contains the same number of points, say z, for each color $i = 1, 2, \ldots, n$, and contains exactly ℓz points of the special color. For $z = 2$, to cover 2ℓ special colored points only, we need all 2ℓ intervals. Hence, for any *fair* covering, we get $z < 2$. This implies that for the covering $\mathcal{C}$, $z = 1$ to meet the overall covering requirement. Since, we need ℓ intervals to cover ℓ special colored points, it is easy to see that the remaining ℓ intervals cover exactly one point of every other color. Hence, the intervals covered corresponds to an EXACT COVER. ■

In the reduced instance above, the number of intervals is dependent only upon the size of the EXACT COVER (ℓ). The EXACT COVER problem is known to be W[1]-hard parameterized by ℓ [83]. Hence, the analogous results for the fair covering problem is summarized as follows:

Theorem 35 *The fair covering problem is* W[1]*-hard parameterized by the number of covering balls* (k)*.*

In dimensions $d \geq 2$, the maximum coverage problem is NP-hard [19], and W[1]-hard [84], *even without the fairness constraint.*

7.3 PTAS using Shifting Technique

In this section, we describe a PTAS for the approximately fair covering problem in any fixed dimension d. Specifically, given an approximate factor $\varepsilon \in [0, 1]$, we want to compute an ε-fair covering of P (using at most k balls in $\mathcal{B}$) such that the number of points covered is at least $(1 - \varepsilon) \cdot \mathsf{opt}$, where opt is the size of an optimal fair covering of P. In other words, the approximation is bi-criteria: one criterion is on the fairness of covering while the other one is on the quality of the solution (i.e., the number of the points covered). We first describe our algorithm for $p = 1$ (a disjoint cover), and later generalize our approach for any constant p. For the simplicity of exposition, we describe the algorithm in two dimensions ($d = 2$) and for two colors ($t = 2$). The extension to higher dimensions and the general case of $t > 2$ colors is straightforward.

7.3.1 Shifted Partitions & Approximate Covering

When solving the fair covering problem in $\mathbb{R}^1$, we were able to compute an optimal (r, b)-covering for any (r, b) pair. This seems quite difficult in higher dimensions, and so we resort to solving an approximate version of this problem as follows. We want to compute a table $\Gamma[1 \ldots n, 1 \ldots n]$ of integers such that for each pair (r, b), we have the following:

1. $\Gamma[r, b]$ is at least the size of an optimal (r, b)-covering, and
2. there exists $r^* \in [(1-\varepsilon)r, r]$ and $b^* \in [(1-\varepsilon)b, b]$ such that $\Gamma[r^*, b^*]$ is at most the size of an optimal (r, b)-covering.

For convenience, we call such a table Γ an *ε-approximate covering table* (ε-ACT) for the instance $(P, \mathcal{B})$. Note that to solve the approximately fair covering problem, it suffices to compute an ε-ACT.

Lemma 44 *Given an ε-ACT Γ for $(P, \mathcal{B})$, one can solve the approximately fair covering problem in polynomial time.*

Proof: Suppose an optimal fair covering covers r_0 red points and b_0 blue points. We call a pair (r, b) with $r, b \in \{1, \ldots, n\}$ *feasible* if **(1)** an (r, b)-covering is fair and **(2)** there exists $r^* \in [(1-\varepsilon)r, r]$ and $b^* \in [(1-\varepsilon)b, b]$ such that $\Gamma[r^*, b^*] \leq k$. We compute all feasible pairs, which can clearly be done in polynomial time given Γ, and find the feasible pair (r, b) that maximizes $r + b$. By definition, we can find $r^* \in [(1-\varepsilon)r, r]$

and $b^* \in [(1-\varepsilon)b, b]$ such that $\Gamma[r^*, b^*] \leq k$. Note that an (r^*, b^*)-covering is ε-fair. Furthermore, $r + b \geq \mathsf{opt}$ since (r_0, b_0) is feasible, hence $r^* + b^* \geq (1-\varepsilon) \cdot \mathsf{opt}$. Because Γ is an ε-ACT, there exists an (r^*, b^*)-covering using at most k (disjoint) disks in $\mathcal{B}$. Therefore, $r^* + b^*$ is a $(1-\varepsilon)$-approximate solution for the approximately fair covering problem. ■

In order to compute an ε-ACT Γ, we use the shifting technique [17]. Let $h = h(\varepsilon)$ be an integer parameter to be determined later. For an integer $i \in \mathbb{Z}$, let $\square_{i,j}$ denote the $h \times h$ square $[i, i+h] \times [j, j+h]$; we say $\square_{i,j}$ is *nonempty* if it contains at least one point in P. We first compute the index set $I = \{(i,j) : \square_{i,j} \text{ is nonempty}\}$. This can be easily done in time polynomial in n and h, by computing for each $p \in P$, the $\mathcal{O}(h^2)$ squares $\square_{i,j}$ that contains p. For each $(i,j) \in I$, define $P_{i,j} = P \cap \square_{i,j}$ and $\mathcal{B}_{i,j} = \{B \in \mathcal{B} : B \subseteq \square_{i,j}\}$. In the next step, we compute a 0-ACT $\Gamma_{i,j}$ for each $(P_{i,j}, \mathcal{B}_{i,j})$ with $(i,j) \in I$. We will show later in Section 7.3.2 how to compute $\Gamma_{i,j}$ in $(n_{i,j}+m_{i,j})^{O(h^2)}$ time, where $n_{i,j} = |P_{i,j}|$ and $m_{i,j} = |\mathcal{B}_{i,j}|$. At this point, let us assume we have the 0-ACTs $\Gamma_{i,j}$ and finish the description of our PTAS. We have the following key observation.

Lemma 45 *Let $\{P_1, \ldots, P_s\}$ be a partition of P and $\mathcal{B}_1, \ldots, \mathcal{B}_s \subseteq \mathcal{B}$ be disjoint subsets such that the disks in $\mathcal{B}_i$ do not cover any points in $P \backslash P_i$. Given 0-ACTs for $(P_1, \mathcal{B}_1), \ldots, (P_s, \mathcal{B}_s)$, we can compute a 0-ACT for $(P, \bigcup_{i=1}^s \mathcal{B}_i)$ in polynomial time.*

Proof: Computing a 0-ACT for $(P, \bigcup_{i=1}^s \mathcal{B}_i)$ is equivalent to computing for all pairs (r, b) the size of the smallest (r, b)-covering of $(P, \bigcup_{i=1}^s \mathcal{B}_i)$. Since the disks in $\mathcal{B}_i$ can only cover the points in P_i, the entire problem instance can be divided into independent sub-problems $(P_1, \mathcal{B}_1), \ldots, (P_s, \mathcal{B}_s)$. This allows us to solve the problem in polynomial time using dynamic programming; see Algorithm 6. ■

For $x, y \in \{0, \ldots, h-1\}$, let $L_{x,y}$ be the set of all integer pairs (i,j) such that $i \bmod h = x$ and $j \bmod h = y$ (See Figure 7.2a). We write $I_{x,y} = I \cap L_{x,y}$.

Algorithm 6 Computing the 0-ACT

Input: $\mathit{\Gamma}_1, \ldots, \mathit{\Gamma}_s$, where $\mathit{\Gamma}_i$ is a 0-ACT for $(P_i, \mathcal{B}_i)$

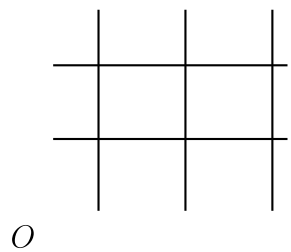
Initialize a $s \times n \times n$ table F with value ∞

for $t \in \{1, \ldots, s\}$; $r, b \in \{1, \ldots, n\}$ **do**

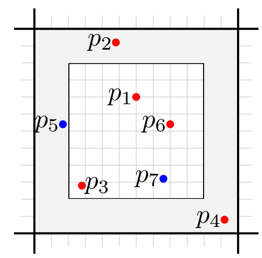
$\quad F[t, r, b] \leftarrow \min_{\substack{0 \le r' \le r \\ 0 \le b' \le b}} \{\mathit{\Gamma}_t[r', b'] + F[t-1, r-r', b-b']\}$

$\mathit{\Gamma}^*[r, b] = F[s, r, b]$ for all $r, b \in \{1, \ldots, n\}$.

return $\mathit{\Gamma}^*$



(a)



(b)

Figure 7.2: (a) The squares $\square_{i,j}$ for $(i, j) \in L_{1,0}$, with $h = 2$. (b) An illustration of the boundary points. The outer square is $\square_{i,j}$ and the inner square is $[i+2, i+h-2] \times [j+2, j+h-2]$, with $h = 12$. The points in the gray region (i.e., p_2, p_4, p_5) are the boundary points in $\square_{i,j}$.

Lemma 46 *For all $x, y \in \{0, \ldots, h-1\}$, the squares $\square_{i,j}$ for $(i, j) \in I_{x,y}$ are interior-disjoint and cover all points in P.*

Proof: Note that the squares $\square_{i,j}$ for $(i, j) \in L_{x,y}$ are interior-disjoint and cover the entire plane $\mathbb{R}^2$ (see Figure 7.2a for an example). It directly follows that the squares $\square_{i,j}$ for $(i, j) \in I_{x,y}$ are interior-disjoint. Consider a point $p \in P$ and let $(i, j) \in L_{x,y}$ such that $p \in \square_{i,j}$. Clearly, $(i, j) \in I$ as $\square_{i,j}$ is nonempty and hence $(i, j) \in I_{x,y}$. Therefore, all points in P are covered by the squares $\square_{i,j}$ for $(i, j) \in I_{x,y}$. ■

Fix $x, y \in \{0, \ldots, h-1\}$. We know by Lemma 46 that $\{P_{i,j} : (i, j) \in I_{x,y}\}$ is a partition of P and the collections $\mathcal{B}_{i,j}$ for $(i, j) \in I_{x,y}$ are disjoint. Furthermore, the disks in $\mathcal{B}_{i,j}$ do not cover any point in $P \backslash P_{i,j}$. Therefore, we can apply Lemma 45 to compute a 0-ACT $\mathit{\Gamma}^{(x,y)}$ for $(P, \bigcup_{(i,j) \in I_{x,y}} \mathcal{B}_{i,j})$ in polynomial time. We do this for all $x, y \in \{0, \ldots, h-1\}$.

Finally, we construct the table $\mathit{\Gamma}$ by setting $\mathit{\Gamma}[r,b] = \min_{x,y\in\{0,\dots,h-1\}} \mathit{\Gamma}^{(x,y)}[r,b]$. We shall show that $\mathit{\Gamma}$ is a $\frac{12h-12}{h^2}$-ACT for $(P,\mathcal{B})$. To this end, we introduce some notions. For a point $p \in P$ and a square $\square_{i,j}$, we say p is a *boundary point* in $\square_{i,j}$ if $p \in \square_{i,j}$ and $p \notin [i+2, i+h-2] \times [j+2, j+h-2]$ (See Figure 7.2b). Now consider some $x, y \in \{0,\dots,h-1\}$. We say $p \in P$ *conflicts* with the pair (x,y) if p is a boundary point in $\square_{i,j}$ where $(i,j) \in I_{x,y}$ is the (unique) pair such that $p \in \square_{i,j}$. One can easily see that each point $p \in P$ conflicts with exactly $h^2 - (h-2)^2$ pairs (x,y).

Lemma 47 *For any $P' \subseteq P$, there exists some $x, y \in \{0,\dots,h-1\}$ such that the number of red (resp., blue) points in P' conflicting with (x,y) is at most $\frac{12h-12}{h^2} \cdot n'_{red}$ (resp., $\frac{12h-12}{h^2} \cdot n'_{blue}$), where n'_{red} (resp., n'_{blue}) is the total number of red (blue) points in P'.*

Proof: Define $\delta^{\text{red}}_{x,y}$ (resp., $\delta^{\text{blue}}_{x,y}$) as the number of the red (resp., blue) points in P' that conflict with (x,y). Because any point $p \in P$ conflicts with exactly $h^2 - (h-2)^2$ pairs (x,y), we have

$$\sum_{x=0}^{h-1}\sum_{y=0}^{h-1} \delta^{\text{red}}_{x,y} = n'_{\text{red}}(h^2 - (h-2)^2) = n'_{\text{red}}(4h-4).$$

Therefore, the number of the pairs (x,y) such that $\delta^{\text{red}}_{x,y} \geq 3n'_{\text{red}}(4h-4)/h^2$ is at most $h^2/3$. Equivalently, the number of the pairs (x,y) such that $\delta^{\text{red}}_{x,y} < 3n'_{\text{red}}(4h-4)/h^2$ is at least $2h^2/3$. For the same reason, the number of the pairs (x,y) such that $\delta^{\text{blue}}_{x,y} < 3n'_{\text{blue}}(4h-4)/h^2$ is at least $2h^2/3$. Since $2h^2/3 + 2h^2/3 > h^2$, there exists at least one pair (x,y) that simultaneously satisfies $\delta^{\text{red}}_{x,y} < 3n'_{\text{red}}(4h-4)/h^2$ and $\delta^{\text{blue}}_{x,y} < 3n'_{\text{blue}}(4h-4)/h^2$. This completes the proof of the lemma. ■

Now we are ready to prove that $\mathit{\Gamma}$ is a $\frac{12h-12}{h^2}$-ACT.

Lemma 48 *$\mathit{\Gamma}$ is a $\frac{12h-12}{h^2}$-ACT for $(P,\mathcal{B})$.*

Proof: Set $\eta = \frac{12h-12}{h^2}$. By the definition of a η-ACT, we have to verify that **(1)** $\Gamma[r,b]$ is at least the size of a smallest (r,b)-covering of $(P,\mathcal{B})$ and **(2)** there exist $r^* \in [(1-\eta)r, r]$ and $b^* \in [(1-\eta)b, b]$ such that $\Gamma[r^*,b^*]$ is at most the size of a smallest (r,b)-covering of $(P,\mathcal{B})$. Condition **(1)** is clearly true. Indeed, for all $x,y \in \{0,\dots,h-1\}$, $\Gamma^{(x,y)}[r,b]$ is the size of the smallest (r,b)-covering of $(P, \bigcup_{(i,j)\in I_{x,y}} \mathcal{B}_{i,j})$ and hence is at least the size of a smallest (r,b)-covering of $(P,\mathcal{B})$. Next, we verify condition **(2)**. Let $\mathcal{B}' \subseteq \mathcal{B}$ be a smallest (r,b)-covering of $(P,\mathcal{B})$ and $P' \subseteq P$ be the points covered by the disks in $\mathcal{B}'$ (hence P' consists of r red points and b blue points). By Lemma 47, there exist $x,y \in \{0,\dots,h-1\}$ such that the number of red (resp., blue) points in P' conflicting with (x,y) is at most ηr (resp., ηb). Let $\mathcal{B}'' = \mathcal{B}' \cap (\bigcup_{(i,j)\in I_{x,y}} \mathcal{B}_{i,j})$ and $P'' \subseteq P'$ be the points covered by the disks in $\mathcal{B}''$. Suppose P'' consists of r^* red points and b^* blue points. Note that any disk in $\mathcal{B}'\backslash\mathcal{B}''$ can only cover the points in P that conflict with (x,y). Therefore, any point in P' that does not conflict with (x,y) must be contained in P'', which implies that $r^* \in [(1-\eta)r, r]$ and $b^* \in [(1-\eta)b, b]$. Since $\Gamma^{(x,y)}$ is a 0-ACT for $(P, \bigcup_{(i,j)\in I_{x,y}} \mathcal{B}_{i,j})$, we have $\Gamma^{(x,y)}[r^*,b^*] \leq |\mathcal{B}''| \leq |\mathcal{B}'|$. It follows that condition **(2)** is also true. ■

We set h to be the smallest integer such that $\frac{12h-12}{h^2} \leq \varepsilon$; clearly, $h = \mathcal{O}(1/\varepsilon)$. Then by the above lemma, Γ is an ε-ACT for $(P,\mathcal{B})$. In this way, we obtain a PTAS for the fair covering problem in $\mathbb{R}^2$.

Theorem 36 *There exists a $(1-\varepsilon)$-approximation algorithm for the fair covering problem in $\mathbb{R}^2$ which runs in $n^{\mathcal{O}(1)} m^{\mathcal{O}(1/\varepsilon^2)}$ time.*

Proof: In our algorithm, the most time-consuming work is the computation of each $\Gamma_{i,j}$ for $(i,j) \in I$, which takes $n_{i,j}^{\mathcal{O}(1)} m_{i,j}^{\mathcal{O}(h^2)}$ time as claimed before. All the other work can be done in time polynomial in h, n, m. Since $I = \mathcal{O}(h^2 n)$, the overall time complexity of our algorithm is $(n+m)^{\mathcal{O}(h^2)}$, i.e., $n^{\mathcal{O}(1)} m^{\mathcal{O}(1/\varepsilon^2)}$. ■

The algorithm can be straightforwardly generalized to higher dimensions and the case $t > 2$, resulting in the following theorem.

Theorem 37 *There exists a* $(1-\varepsilon)$*-approximation algorithm for the* t*-color fair covering problem in* $\mathbb{R}^d$ *which runs in* $n^{\mathcal{O}(t)}m^{\mathcal{O}(1/\varepsilon^d)}$ *time.*

7.3.2 Computing the 0-ACTs $\Gamma_{i,j}$

We now discuss the only missing piece in our algorithm above: the computation of the tables $\Gamma_{i,j}$. Recall that $\Gamma_{i,j}$ is a 0-ACT for $(P_{i,j}, \mathcal{B}_{i,j})$. We show that each $\Gamma_{i,j}$ can be computed in $n_{i,j}^{\mathcal{O}(1)} m_{i,j}^{\mathcal{O}(h^2)}$ time where $n_{i,j} = |P_{i,j}|$ and $m_{i,j} = |\mathcal{B}_{i,j}|$. The key observation is the following.

Lemma 49 *For* $r, b \in \{1, \dots, n_{i,j}\}$*, an* (r, b)*-covering of* $(P_{i,j}, \mathcal{B}_{i,j})$ *is of size at most* $\lfloor h^2/\pi \rfloor$*.*

Proof: Recall that an (r, b)-covering of $(P_{i,j}, \mathcal{B}_{i,j})$ consists of *disjoint* disks in $\mathcal{B}_{i,j}$. All disks in $\mathcal{B}_{i,j}$ are contained in the $h \times h$ square $\Box_{i,j}$. The area of $\Box_{i,j}$ is h^2 and the area of a unit-disk is π. Therefore, any subset of disjoint disks in $\Box_{i,j}$ is of size at most $\lfloor h^2/\pi \rfloor$. ■

With the above observation, we can compute $\Gamma_{i,j}$ as follows. We enumerate all subsets of $\mathcal{B}_{i,j}$ of size at most $\lfloor h^2/\pi \rfloor$, and keep the ones that consist of disjoint disks. In this way, we obtain all (r, b)-coverings of $(P_{i,j}, \mathcal{B}_{i,j})$ for all $r, b \in \{1, \dots, n_{i,j}\}$. By checking these coverings one by one, we can find the smallest (r, b)-covering for all $r, b \in \{1, \dots, n_{i,j}\}$, and hence compute $\Gamma_{i,j}$. The total time cost is $n_{i,j}^{\mathcal{O}(1)} m_{i,j}^{\mathcal{O}(h^2)}$.

7.3.3 Extension to bounded ply

In this section, we give an extension of our algorithm when each point in the plane can be covered by at most a constant number p of disks (also known as constant ply) instead of *disjoint* disks. Our overall approach remains the same as the procedure described in the Subsection 7.3.1, i.e., to solve the approximately fair covering problem, we compute ε-*ACT* Γ from h^2 partitions of the plane. Note that in the previously described approach to obtain 0-*ACTs* $\Gamma_{i,j}$, in other words to solve (r,b)-covering problem for all values of r,b exactly in a constant sized square, we used the enumeration algorithm from the Subsection 7.3.2 which returns *disjoint* coverings. However, we now want to compute coverings for 0-*ACTs* Γ with ply at most p. In other words, we need to solve the (r,b)-covering problem exactly in a constant sized square for coverings with ply at most p. To this end, we show the following lemma analogous to Lemma 49:

Lemma 50 *For $r,b \in \{1,\ldots,n_{i,j}\}$, an (r,b)-covering of $(P_{i,j},\mathcal{B}_{i,j})$ with ply at most p is of size at most ph^2.*

Proof: Note that we want an (r,b)-covering of $(P_{i,j},\mathcal{B}_{i,j})$ consists of subset of disks from $\mathcal{B}_{i,j}$ with ply at most p. Furthermore, all the disks in $\mathcal{B}_{i,j}$ are contained in the $h \times h$ square $\square_{i,j}$.

Consider a unit square s in $\square_{i,j}$. Observe that the distance of any point in s from its midpoint is at most $1/\sqrt{2}$; in particular, the distance is strictly less than 1. Hence, a unit radius disk with its center inside s will cover the midpoint of s. Since we are looking for coverings with ply at most p, s can contain at most p disk centers. Note that $\square_{i,j}$ can be partitioned into h^2 disjoint unit squares, hence, an (r,b)-covering of $(P_{i,j},\mathcal{B}_{i,j})$ with ply at most p can be of size at most ph^2. ∎

Using lem:constant-sized-square, we can compute $\Gamma_{i,j}$ as follows. We enumerate all subsets of $\mathcal{B}_{i,j}$ of size at most ph^2. For each such subset, we compute its ply in $(ph)^{\mathcal{O}(1)}$ time by computing the arrangement of all disks in the subset. In this way, we can obtain all (r,b)-coverings of $(P_{i,j},B_{i,j})$ with ply at most p for all $r,b \in \{1,\ldots,n_{i,j}\}$. By checking these coverings one by one, we can find the smallest (r,b)-covering for all

$r, b \in \{1, \ldots, n_{i,j}\}$. Hence, for computing $\Gamma_{i,j}$, the total time complexity is $n_{i,j}^{\mathcal{O}(1)} m_{i,j}^{\mathcal{O}(ph^2)}$, and for constant p the time complexity is $n_{i,j}^{\mathcal{O}(1)} m_{i,j}^{\mathcal{O}(h^2)}$. This gives us the following result.

Theorem 38 *For coverings with a constant ply, there exists a $(1-\varepsilon)$-approximation algorithm for the fair covering problem in $\mathbb{R}^2$ which runs in $n^{\mathcal{O}(1)} m^{\mathcal{O}(1/\varepsilon^2)}$ time.*

As before, our algorithm can be generalized straightforwardly to higher dimensions and the case $t > 2$; hence, we obtain the following theorem analogous to our last approximation scheme.

Theorem 39 *There exists a $(1-\varepsilon)$-approximation algorithm for the t-color fair covering problem in $\mathbb{R}^d$ which runs in $n^{\mathcal{O}(t)} m^{\mathcal{O}(1/\varepsilon^d)}$ time.*

We note that the running time when p is unbounded is $n^{\mathcal{O}(t)} m^{\mathcal{O}(p/\varepsilon^d)}$.

7.4 Unconstrained Fair Covering in 1D

Our algorithmic results in previous sections provide an efficient way to compute a fair covering (or an approximate fair covering) with *discreteness* and *bounded ply* constraints. We now describe an efficient algorithm to solve the fair covering problem in 1D without any restrictions. We note that our algorithm in this section is similar to the algorithm from Section 7.1. Recall the fair covering problem without restrictions: Given a set of n points on a line each colored with one of the t colors, and a budget k, we want to find a covering with maximum number of points so that each color is covered in proportion to its size. Notice that in this case the input does not contain the candidate set $\mathcal{B}$ or a bound of ply p. For a thorough description, we refer the reader to the beginning of this chapter.

We now turn to the description of our algorithm. Similar to the previous section, we describe our approach for $t = 2$ (i.e., for a set of red and blue points on a line).

We solve the fair covering problem, again by computing an optimal (r, b)-covering for all $r, b \in \{1, \ldots, n\}$. Without loss of generality, the points $p_1, \ldots, p_n$ are sorted in left-to-right order. For each $i \in \{1, \ldots, n\}$, let $\pi_i < i$ be the largest index such that the distance between p_{π_i} and p_{i+1} is greater than 1 (suppose $p_{n+1} = \infty$); if such an index does not exist, we set $\pi_i = 0$. Since $\pi_1 \leq \cdots \leq \pi_n$, we can compute all π_i in $\mathcal{O}(n)$ time by making a left-to-right pass over the set of input points $p_1, \ldots, p_n$.

Define $F[i, r, b]$ as the size of an optimal (r, b)-covering that covers no points in $P\{p_1, \ldots, p_i\}$. We compute $F[i, r, b]$ for all (i, r, b) tuples. For the pairs (r, b) such that no (r, b)-covering exists, we set $F[i, r, b] = \infty$. We now give a recurrence to computer $F[i, r, b]$.

Claim 5

$$F[i, r, b] \;=\; \min \begin{Bmatrix} F[i-1, r, b], \\ \min\limits_{0 \leq j \leq \pi_i} (\ell_{i,j} + F[j, r - r_{i,j}, b - b_{i,j}]) \end{Bmatrix}$$

where $\ell_{i,j}$ is the minimum number of unit intervals required to cover set of points $\{p_{j+1}, \ldots, p_i\}$, and $r_{i,j}$ (resp., $b_{i,j}$) is the number of red (resp., blue) points covered by these $\ell_{i,j}$ intervals.

The idea of the above recurrence as follows: The first term $F[i-1, r, b]$ considers the case when p_i is not covered. Next, we take a minimum over at most i terms where we consider the case when p_i is covered. We "guess" the rightmost uncovered point. Since we are not allowed to cover p_{i+1}, if p_i is covered, then all points in $\{p_{\pi_i+1}, \ldots, p_i\}$ are covered. Therefore, the rightmost uncovered point must have index smaller than or equal to π_i. The recurrence relation from Claim 5 points us to a dynamic programming procedure shown in Algorithm 7 to efficiently compute the table F. The base cases for the dynamic program is $F[i, 0, 0] = 0$ for all $i \in 0, \ldots, n$ and $F[0, r, b] = \infty$ for all $r, b \in \{1, \ldots, n\}$.

Lemma 51 *Algorithm 7 can be implemented in worst-case time $\mathcal{O}(n^4)$.*

Algorithm 7 Computing the F-table

Input: The set of points P

Compute π_i for $i \in \{1, \dots, n\}$

Initialize n $\times r \times b$ sized table with value ∞

for $i \in \{0, \dots, n\}$; $r, b \in \{0, \dots, n\}$ **do**

$\quad F[i, r, b] \leftarrow \min\{F[i-1, r, b], \min\limits_{0 \leq j \leq \pi_i} (\ell_{i,j} + F[j, r - r_{i,j}, b - b_{i,j}])\}$

return F

Proof: Sorting P takes $\mathcal{O}(n \log n)$ time. Computing π_i for $i \in \{1, \dots, n\}$ takes overall $\mathcal{O}(n)$ time using the monotonicity of π_i (i.e., $\pi_{i+1} \geq \pi_i$). After that the F-table can be computed in $\mathcal{O}(n^4)$ time as there are n^3 entries in the table, and each entry takes $\mathcal{O}(n)$ time for at most n look-ups. Note that $\ell_{i,j}$ can be computed in a constant time given the coordinates of p_{j+1} and p_i, in particular, $\ell_{i,j} = \lceil (p_i - p_{j+1}) \rceil$. ■

We recall that after F-table is computed, we can solve the fair covering problem by checking all entries in the table for which the (r, b)-covering is fair and has $F[n, r, b] \leq k$. Among all such valid pairs, we return the pair (r^*, b^*) with the maximum $r^* + b^*$. Clearly, $c^* = r^* + b^*$ is the optimum of the problem instance. We therefore summarize the main result of this section as follows.

Theorem 40 *The unconstrained fair covering problem in $\mathbb{R}^1$ with $t = 2$ colors can be solved in $\mathcal{O}(n^4)$ time.*

Observe that it is straightforward to generalize our approach for arbitrary number of colors using a dynamic program with a $t + 1$-dimensional DP table. Hence, we obtain the following theorem:

Theorem 41 *The unconstrained fair covering problem in $\mathbb{R}^1$ can be solved in $\mathcal{O}(n^{t+2})$ time.*

7.5 Bibliographic Notes

The problem of covering points by balls or other geometric shapes has a long history in computational geometry, operations research, and theoretical computer science, due to its natural connections to clustering and facility location problems [85, 17, 86, 87, 88]. It is known that covering a set of two-dimensional points with a minimum number of unit disks is NP-hard, and so is the problem of maximizing the number of points covered by k unit disks [19, 84, 89, 90]. The setting of minimum ply coverage for all points in the plane has been studied recently in [91], and minimum ply coverage only of input points is considered in [92] and the following works. Recently, a number of researchers have considered clustering and covering problems with an additional constraint of fairness. In this setting, the input consists of points belonging to different colors (classes), and the goal is to find a solution where *each cluster* has approximately equal representation of all colors [93, 78, 94, 95, 96]. These formulations are different from our model because we allow individual clusters to be unbalanced as long as *in aggregate* each color receives its fair share. This non-local form of fair representation seems much harder than requiring each cluster to locally meet the balance condition. In another line of work, [97, 76, 98] consider a *colorful* variant of the k-center problem where the goal is to satisfy a *minimum* coverage for each color type. The colorful covering however does not achieve fairness because some color classes can have arbitrarily high representation in the output, as long as other colors meet the minimum threshold. In fact, enforcing the fairness by controlling *both* the lower and the upper bounds of representation seems to be a much harder problem, as suggested by some of our hardness results in one dimension.

Chapter 8

Conclusion and Open Problems

In this dissertation, we studied the committee selection problem where the goal is to choose a fixed number of candidates based on voters' preferences. This problem captures many real-life scenarios such as choosing representatives in a democracy, staff hiring, jury selection, etc. Our work investigates four natural committee selection problems dealing with some of the crucial aspects of committee selection such as winner determination, fault tolerance, and fairness. We made progress towards a better understanding of these problems by making clean theoretical formulations for these problems, and designing a family of non-trivial approximation and exact algorithms for them. Although we leave some questions unanswered, this dissertation will hopefully serve as an important first step in the study of these class of problems.

We began the discussion in Chapter 2 with the multiwinner elections in Euclidean space under minimax Chamberlin-Courant voting rules. First, we settle the complexity of the winner determination problem by showing it is NP-hard for dimensions $d \geq 2$, but we followed that up with several (nearly-optimal) approximation bounds which are elusive in the non-Euclidean setting. We believe that our approximation algorithms are robust and will generalize to many other interesting questions, for instance, most of our algorithmic results (except for Theorem 5) extend to the recently studied egalitarian k-median rules [99] when considered for the Euclidean elections.

In our work, we considered the min-max (ℓ_∞ aggregation function. At a quick glance, we believe that our NP-hardness proof from Theorem 1 holds for the ℓ_1-aggregation function as well. Usually, the approximability of min-max and min-sum variants of a problem differs significantly. For example, for the k-center clustering strong inapproximability results are known (even in a Euclidean space) but there are several approximation schemes are known for k-means clustering [100, 101]. Therefore, a natural open problem to consider is the following:

Open Problem 1 (Winner Determination) *Does the winner determination problem admits better approximation bounds under the min-sum (ℓ_1) scoring function?*

Going further in this direction, resolving the complexity and approximation bounds for other important voting rules, such as the utilitarian general class of OWA rules [37, 7] for the Euclidean elections. An interesting avenue to consider is Exploring Euclidean elections when the positions of voters and candidates are known only approximately, for instance, each placed at some unknown point in a disk. Such data uncertainty naturally exists in many real-world applications. Notice that such a setting with uncertainty relates to the necessary and possible winner problems under incomplete preferences [102].

Open Problem 2 (Winner Determination) *What is the complexity of finding a winning committee under the Chamberlin-Courant voting rule for Euclidean elections when the candidate/voter positions are given as a probability distribution in a compact set?*

In Chapters 3 and 4, we introduce a novel fault-tolerance model and study its complexity on Euclidean elections under Chamberlin Courant rule (CC-rule). For one-dimensional instances, we give polynomial-time algorithms to solve these problems and for $d \geq 2$, we show NP-hardness, and present several (greedy based) constant factor approximations and an FPT approximation scheme.

Our work suggests several new and exciting research directions for achieving fault-tolerance in committees, such as (1) extending our results to other commonly used scoring functions such as k-median or k-means [103], (2) Investigating fault tolerance on the ordinal voter preference models for various well-known committee selection rules. Intuitively, it seems that dealing with fault-tolerance is *computationally* easier when an optimal winner computation is polynomial time (we show this with $1d$ and single-peaked elections). It is interesting to further investigate this intuition on other voting scenarios from [31, 33].

Next, in Chapters 5 and 6, we consider (Committee) Winner Verification (WV) and the Candidate Winner (CW) problem, respectively. We settle the complexity of these two problems for Chamberlin-Courant and Monroe voting rule by showing Winner Verification problem to be coNP-complete and Candidate Winner problem to be θ_2^P complete for both rankings and approval ballots. On the positive side, we show that for single-peaked and single-crossing preferences, both the problems can be solved in polynomial time for Chamberlin-Courant rule.

With the backdrop of the above hardness results in the (general) unstructured setting, it is interesting to study these two problems on restricted preference domains. In our work, we only give positives results for the Chamberlin-Courant rule but the following directions are interesting:

Open Problem 3 (Winner Verification Problems) *Are WV and CW polynomial-time solvable under the Monroe voting rule on single-peaked or single-crossing preferences?*

Open Problem 4 (Winner Verification Problems) *What is the complexity of WV and CW for (multidimensional) Euclidean elections under Chamberlin-Courant and Monroe rule?*

We believe that WV and CW will remain NP-hard even in Euclidean elections but we hope that one can obtain polynomial-time algorithms with good approximation guarantees in this case.

Finally, in Chapter 7, we study max covering of (multicolored) points in $\mathbb{R}^d$ using unit radius balls under fairness constraints (in particular, we consider proportional fairness). We proved that the problem is NP-hard even in one dimension when the number of color groups is large. When the number of colors is fixed, we presented a polynomial time exact algorithm in one dimension, and a PTAS in any fixed dimension. Our work suggests many interesting open problems, including whether one can achieve a constant factor approximation significantly faster than our PTAS, and whether the PTAS can be achieved for covering with arbitrary ply.

Bibliography

[1] N. Betzler, A. Slinko, and J. Uhlmann, *On the computation of fully proportional representation*, *Journal of Artificial Intelligence Research* **47** (2013) 475–519. (Cited on pages 2, 4, 16, 46, 48, and 58)

[2] R. Bredereck, A. Kaczmarczyk, and R. Niedermeier, *On coalitional manipulation for multiwinner elections: Shortlisting*, *Autonomous Agents and Multi-Agent Systems* **35** (2021), no. 2 38. (Cited on pages 2 and 48)

[3] E. J. Friedman, V. Gkatzelis, C.-A. Psomas, and S. Shenker, *Fair and efficient memory sharing: Confronting free riders*, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 1965–1972, 7, 2019. (Cited on pages 2 and 48)

[4] A. Goel, A. K. Krishnaswamy, S. Sakshuwong, and T. Aitamurto, *Knapsack voting for participatory budgeting*, *ACM Transactions on Economics and Computation (TEAC)* **7** (2019) 1–27. (Cited on pages 2 and 48)

[5] T. Lu and C. Boutilier, *Budgeted social choice: From consensus to personalized decision making*, in *Twenty-Second International Joint Conference on Artificial Intelligence*, pp. 280–286, 2011. (Cited on pages 2, 4, 16, and 48)

[6] S. Maiyya, V. Zakhary, D. Agrawal, and A. E. Abbadi, *Database and distributed computing fundamentals for scalable, fault-tolerant, and consistent maintenance of blockchains*, *Proceedings of the VLDB Endowment* **11** (2018), no. 12. (Cited on pages 2 and 48)

[7] P. Skowron, P. Faliszewski, and J. Lang, *Finding a collective set of items: From proportional multirepresentation to group recommendation*, *Artificial Intelligence* **241** (2016) 191–216. (Cited on pages 2, 48, and 158)

[8] J. Kleinberg and E. Tardos, *Algorithm design*. Pearson Education India, 2006. (Cited on pages 2 and 17)

[9] M. Sipser, *Introduction to the theory of computation*, *ACM Sigact News* **27** (1996), no. 1 27–29. (Cited on page 2)

[10] S. Arora and B. Barak, *Computational complexity: a modern approach*. Cambridge University Press, 2009. (Cited on page 2)

[11] J. R. Chamberlin and P. N. Courant, *Representative deliberations and representative decisions: Proportional representation and the borda rule*, *American Political Science Review* **77** (1983), no. 3 718–733. (Cited on pages 3, 16, and 93)

[12] T. F. Gonzalez, *Clustering to minimize the maximum intercluster distance*, *Theoretical computer science* **38** (1985) 293–306. (Cited on pages 4, 48, 81, and 104)

[13] L. Yu, H. Chan, and E. Elkind, *Multiwinner elections under preferences that are single-peaked on a tree*, in *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013. (Cited on pages 4 and 16)

[14] P. Skowron, L. Yu, P. Faliszewski, and E. Elkind, *The complexity of fully proportional representation for single-crossing electorates*, *Theoretical Computer Science* **569** (2015) 43–57. (Cited on pages 4, 16, and 46)

[15] E. Elkind, M. Lackner, and D. Peters, *Structured preferences*, *Trends in computational social choice* (2017) 187–207. (Cited on pages 4, 16, and 46)

[16] M. de Berg and A. Khosravi, *Optimal binary space partitions in the plane*, in *International Computing and Combinatorics Conference*, pp. 216–225, Springer, 2010. (Cited on pages 5, 18, and 61)

[17] D. S. Hochbaum and W. Maass, *Approximation schemes for covering and packing problems in image processing and VLSI*, *Journal of the ACM (JACM)* (1985) 130–136. (Cited on pages 8, 11, 83, 147, and 156)

[18] B. L. Monroe, *Fully proportional representation*, *American Political Science Review* **89** (1995), no. 4 925–940. (Cited on pages 9, 93, and 96)

[19] R. J. Fowler, M. S. Paterson, and S. L. Tanimoto, *Optimal packing and covering in the plane are NP-complete*, *Information processing letters* (1981) 133–137. (Cited on pages 11, 145, and 156)

[20] C. Sonar, S. Suri, and J. Xue, *Multiwinner elections under minimax chamberlin-courant rule in euclidean space*, 2022. (Cited on pages 13 and 62)

[21] C. Sonar, S. Suri, and J. Xue, *Fault Tolerance in Euclidean Committee Selection*, in *31st Annual European Symposium on Algorithms (ESA 2023)* (I. L. Gørtz, M. Farach-Colton, S. J. Puglisi, and G. Herman, eds.), vol. 274 of *Leibniz International Proceedings in Informatics (LIPIcs)*, (Dagstuhl, Germany), pp. 95:1–95:14, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. (Cited on page 13)

[22] C. Sonar, P. Dey, and N. Misra, *On the complexity of winner verification and candidate winner for multiwinner voting rules*, in *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pp. 89–95, 2021. (Cited on pages 13 and 90)

[23] D. Lokshtanov, C. Sonar, S. Suri, and J. Xue, *Fair covering of points by balls.*, in *CCCG*, pp. 26–32, 2020. (Cited on page 13)

[24] Ú. Hébert-Johnson, C. Sonar, S. Suri, and V. Surianarayanan, *Anonymity-preserving space partitions*, in *32nd International Symposium on Algorithms and Computation (ISAAC 2021)*, Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021. (Cited on page 13)

[25] N. Misra, C. Sonar, P. Vaidyanathan, and R. Vaish, *Equitable division of a path*, *arXiv preprint arXiv:2101.09794* (2021). (Cited on page 14)

[26] V. Himakunthala, A. Ouyang, D. Rose, R. He, A. Mei, Y. Lu, C. Sonar, M. Saxon, and W. Wang, *Let's think frame by frame with VIP: A video infilling and prediction dataset for evaluating video chain-of-thought*, in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (H. Bouamor, J. Pino, and K. Bali, eds.), (Singapore), pp. 204–219, Association for Computational Linguistics, Dec., 2023. (Cited on page 14)

[27] D. Rose, V. Himakunthala, A. Ouyang, R. He, A. Mei, Y. Lu, M. Saxon, C. Sonar, D. Mirza, and W. Y. Wang, *Visual chain of thought: Bridging logical gaps with multimodal infillings*, *arXiv preprint arXiv:2305.02317* (2023). (Cited on page 14)

[28] D. Peters, *Recognising multidimensional euclidean preferences*, in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 642–648, 2017. (Cited on page 15)

[29] C. D. Toth, J. O'Rourke, and J. E. Goodman, *Handbook of discrete and computational geometry*. CRC press, 2017. (Cited on page 33)

[30] N. H. Mustafa and S. Ray, *Improved results on geometric hitting set problems*, *Discrete & Computational Geometry* **44** (2010) 883–895. (Cited on page 39)

[31] E. Elkind, P. Faliszewski, P. Skowron, and A. Slinko, *Properties of multiwinner voting rules*, *Social Choice and Welfare* **48** (2017), no. 3 599–632. (Cited on pages 46 and 159)

[32] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon, *Multiwinner voting: A new challenge for social choice theory*, *Trends in computational social choice* **74** (2017), no. 2017 27–47. (Cited on pages 46 and 90)

[33] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon, *Committee scoring rules: Axiomatic characterization and hierarchy*, *ACM Transactions on Economics and Computation (TEAC)* **7** (2019), no. 1 1–39. (Cited on pages 46, 90, and 159)

[34] H. Aziz, P. Faliszewski, B. Grofman, A. Slinko, and N. Talmon, *Egalitarian committee scoring rules*, in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pp. 56–62, International Joint Conferences on Artificial Intelligence Organization, 7, 2018. (Cited on page 46)

[35] M. T. Godziszewski, P. Batko, P. Skowron, and P. Faliszewski, *An analysis of approval-based committee rules for 2d-euclidean elections*, *Proceedings of the AAAI Conference on Artificial Intelligence* **35** (2021), no. 6 5448–5455. (Cited on page 46)

[36] K. Munagala, Z. Shen, and K. Wang, *Optimal algorithms for multiwinner elections and the chamberlin-courant rule*, *EC '21: Proceedings of the 22nd ACM Conference on Economics and Computation* (2021) 697–717. (Cited on pages 46 and 90)

[37] P. Skowron, P. Faliszewski, and A. Slinko, *Achieving fully proportional representation: Approximability results*, *Artificial Intelligence* **222** (2015) 67–103. (Cited on pages 46, 90, and 158)

[38] J. Byrka, P. Skowron, and K. Sornat, *Proportional approval voting, harmonic k-median, and negative association*, in *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, vol. 107, p. 26, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018. (Cited on page 46)

[39] O. A. Davis, M. J. Hinich, and P. C. Ordeshook, *An expository development of a mathematical model of the electoral process*, *American political science review* **64** (1970), no. 2 426–448. (Cited on page 47)

[40] K. Arrow, *Advances in the spatial theory of voting.* Cambridge University Press, 1990. (Cited on page 47)

[41] C. Sonar, S. Suri, and J. Xue, *Multiwinner elections under minimax chamberlin-courant rule in euclidean space*, in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22* (L. D. Raedt, ed.), pp. 475–481, International Joint Conferences on Artificial Intelligence Organization, 7, 2022. (Cited on page 47)

[42] N. Misra and C. Sonar, *Robustness radius for chamberlin-courant on restricted domains*, in *International Conference on Current Trends in Theory and Practice of Informatics*, pp. 341–353, Springer, 2019. (Cited on page 58)

[43] N. Misra, C. Sonar, and P. Vaidyanathan, *On the complexity of chamberlin-courant on almost structured profiles*, in *International Conference on Algorithmic Decision Theory*, pp. 124–138, Springer, 2017. (Cited on page 58)

[44] C. Sonar, *Problems in computational social choice on restricted domains*, 2019. (Cited on page 58)

[45] V. Nagarajan, B. Schieber, and H. Shachnai, *The euclidean k-supplier problem*, in *Integer Programming and Combinatorial Optimization: 16th International Conference, IPCO 2013, Valparaíso, Chile, March 18-20, 2013. Proceedings 16*, pp. 290–301, Springer, 2013. (Cited on page 74)

[46] D. S. Hochbaum and D. B. Shmoys, *A unified approach to approximation algorithms for bottleneck problems*, *Journal of the ACM (JACM)* **33** (1986), no. 3 533–550. (Cited on page 81)

[47] E. Elkind, P. Faliszewski, J.-F. Laslier, P. Skowron, A. Slinko, and N. Talmon, *What do multiwinner voting rules do? an experiment over the two-dimensional euclidean domain*, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31 of *AAAI'17*, pp. 494–501, 2017. (Cited on page 90)

[48] T. Lu and C. E. Boutilier, *The unavailable candidate model: a decision-theoretic view of social choice*, in *Proceedings of the 11th ACM conference on Electronic commerce*, pp. 263–274, 2010. (Cited on page 90)

[49] A. Grivet Sébert, N. Maudet, P. Perny, and P. Viappiani, *Preference aggregation in the generalised unavailable candidate model*, in *International Conference on Algorithmic Decision Theory*, pp. 35–50, Springer, 2021. (Cited on page 90)

[50] P. Faliszewski, E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe, *Llull and copeland voting computationally resist bribery and constructive control*, *J. Artif. Intell. Res.* **35** (2009) 275–341. (Cited on page 90)

[51] E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe, *Anyone but him: The complexity of precluding an alternative*, *Artificial Intelligence* **171** (2007), no. 5-6 255–285. (Cited on page 90)

[52] S. Chaudhuri, N. Garg, and R. Ravi, *The p-neighbor k-center problem*, *Information Processing Letters* **65** (1998), no. 3 131–134. (Cited on page 90)

[53] S. O. Krumke, *On a generalization of the p-center problem*, *Information processing letters* **56** (1995), no. 2 67–71. (Cited on page 90)

[54] S. Khuller, R. Pless, and Y. J. Sussmann, *Fault tolerant k-center problems*, *Theoretical Computer Science* **242** (2000), no. 1-2 237–245. (Cited on page 90)

[55] C. Swamy and D. B. Shmoys, *Fault-tolerant facility location*, *ACM Transactions on Algorithms (TALG)* **4** (2008), no. 4 1–27. (Cited on page 90)

[56] M. Hajiaghayi, W. Hu, J. Li, S. Li, and B. Saha, *A constant factor approximation algorithm for fault-tolerant k-median*, *ACM Transactions on Algorithms (TALG)* **12** (2016), no. 3 1–19. (Cited on page 90)

[57] T. Lu and C. Boutilier, *Budgeted social choice: From consensus to personalized decision making*, in *Twenty-Second International Joint Conference on Artificial Intelligence*, pp. 280–286, 2011. (Cited on page 93)

[58] A. D. Procaccia, J. S. Rosenschein, and A. Zohar, *On the complexity of achieving proportional representation*, *Social Choice and Welfare* **30** (2008) 353–362. (Cited on pages 93 and 104)

[59] R. Bredereck, P. Faliszewski, A. Kaczmarczyk, R. Niedermeier, P. Skowron, and N. Talmon, *Robustness among multiwinner voting rules*, *CoRR* **abs/1707.01417** (2017) [arXiv:1707.0141]. (Cited on pages 93 and 115)

[60] R. Bredereck, P. Faliszewski, A. Kaczmarczyk, R. Niedermeier, P. Skowron, and N. Talmon, *Robustness among multiwinner voting rules*, in *10th International Symposium of Algorithmic Game Theory (SAGT)*, pp. 80–92, 2017. (Cited on pages 93, 115, and 116)

[61] C. L. Dodgson, *A method of taking votes on more than two issues*, in *Classics of Social Choice*, pp. 288–297, University of Michigan Press (Reprinted in 1995), 1876. (Cited on pages 94 and 114)

[62] E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe, *Exact analysis of dodgson elections: Lewis carroll's 1876 voting system is complete for parallel access to np*, *Journal of the ACM (JACM)* (1997) 806–825. (Cited on page 94)

[63] M. Lackner and P. Skowron, *Consistent approval-based multi-winner rules*, in *Proceedings of the 2018 ACM Conference on Economics and Computation*, pp. 47–48, ACM, 2018. (Cited on page 96)

[64] C. H. Papadimitriou and S. K. Zachos, *Two remarks on the power of counting*, in *Theoretical Computer Science*, pp. 269–275. Springer, 1982. (Cited on pages 97 and 114)

[65] K. W. Wagner, *Bounded query classes*, *SIAM Journal on Computing* **19** (1990), no. 5 833–846. (Cited on page 97)

[66] L. A. Hemachandra, *The strong exponential hierarchy collapses*, *Journal of Computer and System Sciences* **39** (1989), no. 3 299–322. (Cited on page 97)

[67] E. Hemaspaandra, H. Spakowski, and J. Vogel, *The complexity of Kemeny elections*, *Theoretical Computer Science* **349** (2005), no. 3 382–391. (Cited on pages 97 and 114)

[68] N. Betzler, A. Slinko, and J. Uhlmann, *On the computation of fully proportional representation*, *Journal of Artificial Intelligence Research* **47** (2013) 475–519. (Cited on pages 104, 111, and 134)

[69] P. Skowron, L. Yu, P. Faliszewski, and E. Elkind, *The complexity of fully proportional representation for single-crossing electorates*, *Theoretical Computer Science* **569** (2015) 43–57. (Cited on pages 111, 132, 133, and 134)

[70] P. Skowron, L. Yu, P. Faliszewski, and E. Elkind, *The complexity of fully proportional representation for single-crossing electorates*, in *International Symposium on Algorithmic Game Theory*, pp. 1–12, Springer, 2013. (Cited on page 111)

[71] J. Bartholdi, C. A. Tovey, and M. A. Trick, *Voting schemes for which it can be difficult to tell who won the election*, *Social Choice and welfare* **6** (1989), no. 2 157–165. (Cited on page 114)

[72] J. G. Kemeny, *Mathematics without numbers*, *Daedalus* **88** (1959), no. 4 577–591. (Cited on page 114)

[73] J. Kleinberg, H. Lakkaraju, J. Leskovec, J. Ludwig, and S. Mullainathan, *Human decisions and machine predictions*, *The quarterly journal of economics* **133** (2018), no. 1 237–293. (Cited on page 136)

[74] M. B. Zafar, I. Valera, M. Rodriguez, K. Gummadi, and A. Weller, *From parity to preference-based notions of fairness in classification*, in *Advances in Neural Information Processing Systems*, pp. 229–239, 2017. (Cited on page 136)

[75] X. Chen, B. Fain, L. Lyu, and K. Munagala, *Proportionally Fair Clustering*, in *International Conference on Machine Learning*, pp. 1032–1041, 2019. (Cited on page 136)

[76] X. Jia, K. Sheth, and O. Svensson, *Fair Colorful k-Center Clustering*, in *Integer Programming and Combinatorial Optimization*, pp. 209–222, Springer, 2020. (Cited on pages 136 and 156)

[77] A. Backurs, P. Indyk, K. Onak, B. Schieber, A. Vakilian, and T. Wagner, *Scalable fair clustering*, *arXiv preprint arXiv:1902.03519* (2019). (Cited on page 136)

[78] F. Chierichetti, R. Kumar, S. Lattanzi, and S. Vassilvitskii, *Fair clustering through fairlets*, in *Advances in Neural Information Processing Systems*, pp. 5029–5037, 2017. (Cited on pages 136 and 156)

[79] M. Kleindessner, P. Awasthi, and J. Morgenstern, *Fair k-Center Clustering for Data Summarization*, in *International Conference on Machine Learning*, pp. 3448–3457, 2019. (Cited on page 136)

[80] L. E. Celis, V. Keswani, D. Straszak, A. Deshpande, T. Kathuria, and N. K. Vishnoi, *Fair and diverse DPP-based data summarization*, *arXiv preprint arXiv:1802.04023* (2018). (Cited on page 136)

[81] S. C. Nandy, S. Pandit, and S. Roy, *Covering points: Minimizing the maximum depth*, in *CCCG*, pp. 37–42, 2017. (Cited on page 140)

[82] R. M. Karp, *Reducibility among combinatorial problems*, in *Complexity of computer computations*, pp. 85–103. Springer, 1972. (Cited on page 143)

[83] R. G. Downey and M. R. Fellows, *Fixed-parameter tractability and completeness ii: On completeness for W[1]*, *Theoretical Computer Science* **141** (1995), no. 1-2 109–131. (Cited on page 145)

[84] D. Marx, *Efficient approximation schemes for geometric problems?*, in *European Symposium on Algorithms*, pp. 448–459, 2005. (Cited on pages 145 and 156)

[85] E. M. Arkin and R. Hassin, *Approximation algorithms for the geometric covering salesman problem*, *Discrete Applied Mathematics* **55** (1994), no. 3 197–218. (Cited on page 156)

[86] N. Megiddo and K. J. Supowit, *On the complexity of some common geometric location problems*, *SIAM journal on computing* **13** (1984), no. 1 182–196. (Cited on page 156)

[87] C. Toregas, R. Swain, C. ReVelle, and L. Bergman, *The location of emergency service facilities*, *Operations research* **19** (1971), no. 6 1363–1373. (Cited on page 156)

[88] D. Yang, S. Misra, X. Fang, G. Xue, and J. Zhang, *Two-tiered constrained relay node placement in wireless sensor networks: Computational complexity and efficient approximations*, *IEEE Transactions on Mobile Computing* **11** (2011), no. 8 1399–1411. (Cited on page 156)

[89] T. M. Chan and E. Grant, *Exact algorithms and APX-hardness results for geometric packing and covering problems*, *Computational Geometry* **47** (2014), no. 2 112–124. (Cited on page 156)

[90] K. L. Clarkson and K. Varadarajan, *Improved approximation algorithms for geometric set cover*, *Discrete & Computational Geometry* **37** (2007), no. 1 43–58. (Cited on page 156)

[91] T. Biedl, A. Biniaz, and A. Lubiw, *Minimum ply covering of points with disks and squares*, *arXiv preprint arXiv:1905.00790* (2019). (Cited on page 156)

[92] T. Erlebach and E. J. van Leeuwen, *Approximating geometric coverage problems*, in *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1267–1276, 2008. (Cited on page 156)

[93] C. Rösner and M. Schmidt, *Privacy Preserving Clustering with Constraints*, in *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018. (Cited on page 156)

[94] S. Bera, D. Chakrabarty, N. Flores, and M. Negahbani, *Fair algorithms for clustering*, in *Advances in Neural Information Processing Systems*, pp. 4955–4966, 2019. (Cited on page 156)

[95] S. Ahmadian, A. Epasto, R. Kumar, and M. Mahdian, *Clustering without over-representation*, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 267–275, 2019. (Cited on page 156)

[96] M. Schmidt, C. Schwiegelshohn, and C. Sohler, *Fair coresets and streaming algorithms for fair k-means*, in *International Workshop on Approximation and Online Algorithms*, pp. 232–251, Springer, 2019. (Cited on page 156)

[97] S. Bandyapadhyay, T. Inamdar, S. Pai, and K. Varadarajan, *A Constant Approximation for Colorful k-Center*, in *27th Annual European Symposium on Algorithms (ESA 2019)*, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019. (Cited on page 156)

[98] G. Anegg, H. Angelidakis, A. Kurpisz, and R. Zenklusen, *A Technique for Obtaining True Approximations for k-Center with Covering Constraints*, in *International Conference on Integer Programming and Combinatorial Optimization*, pp. 52–65, Springer, 2020. (Cited on page 156)

[99] S. Gupta, P. Jain, S. Saurabh, and N. Talmon, *Even more effort towards improved bounds and fixed-parameter tractability for multiwinner rules*, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)* (2021). (Cited on page 158)

[100] J. Matoušek, *On approximate geometric k-clustering*, *Discrete & Computational Geometry* **24** (2000), no. 1 61–84. (Cited on page 158)

[101] D. Feldman, M. Monemizadeh, and C. Sohler, *A ptas for k-means clustering based on weak coresets*, in *Proceedings of the twenty-third annual symposium on Computational geometry*, pp. 11–18, 2007. (Cited on page 158)

[102] S. Gaspers, V. Naroditskiy, N. Narodytska, and T. Walsh, *Possible and necessary winner problem in social polls*, *arXiv preprint arXiv:1302.1669* (2013). (Cited on page 158)

[103] A. K. Jain and R. C. Dubes, *Algorithms for clustering data.* Prentice-Hall, Inc., 1988. (Cited on page 159)