Algorithm Design for Ordinal Settings

by

Haripriya Pulyassary

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2022

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Social choice theory is concerned with aggregating the preferences of agents into a single outcome. While it is natural to assume that agents have cardinal utilities, in many contexts, we can only assume access to the agents' ordinal preferences, or rankings over the outcomes. As ordinal preferences are not as expressive as cardinal utilities, a loss of efficiency is unavoidable. Procaccia and Rosenschein [39] introduced the notion of *distortion* to quantify this worst-case efficiency loss for a given social choice function.

We primarily study distortion in the context of election, or equivalently clustering problems, where we are given a set of agents and candidates in a metric space; each agent has a preference ranking over the set of candidates and we wish to elect a committee of k candidates that minimizes the total social cost incurred by the agents.

In the single-winner setting (when $k = 1$), we give a novel LP-duality based analysis framework that makes it easier to analyze the distortion of existing social choice functions, and extends readily to randomized social choice functions. Using this framework, we show that it is possible to give simpler proofs of known results. We also show how to efficiently compute an *optimal* randomized social choice function for any given instance. We utilize the latter result to obtain an instance for which any randomized social choice function has distortion at least 2.063164. This disproves the long-standing conjecture that there exists a randomized social choice function that has a worst-case distortion of at most 2.

When $k \geq 2$, it is not possible to compute an $O(1)$-distortion committee using purely ordinal information. We develop two $O(1)$-distortion mechanisms for this problem: one having a $polylog(n)$ (per agent) query complexity, where n is the number of agents; and the other having $O(k)$ query complexity (i.e., no dependence on n). We also study a much more general setting called *minimum-norm k-clustering* recently proposed in the clustering literature, where the objective is some monotone, symmetric norm of the the agents' costs, and we wish to find a committee of k candidates to minimize this objective. When the norm is the sum of the ℓ largest costs, which is called the *ℓ-centrum problem* in the clustering literature, we give low-distortion mechanisms by adapting our mechanisms for k-median. En route, we give a simple adaptive-sampling algorithm for this problem. Finally, we show

how to leverage this adaptive-sampling idea to also obtain a constant-factor bicriteria approximation algorithm for minimum-norm k-clustering (in its full generality).

Acknowledgements

First and foremost, I thank my supervisor, Professor Chaitanya Swamy. This thesis certainly would not have been possible without his guidance, support, insights, encouragement, and in general, his very hands-on approach to advising. I am grateful to him for his extremely thorough and detailed comments on all of my work, as well as for all of the advice that he has given me. I could not have asked for a better advisor. My debt to Swamy is a bit deeper than just the past year, as it was his undergrad discrete optimization course that piqued my interest in this area, and convinced me to give algorithms a second chance. Thank you, Swamy.

I thank my readers, Professor Jochen Koenemann and Professor Kate Larson, for reading my thesis and providing valuable comments, and more generally, for the lessons that they have taught me as a URA, and the encouragement and support that they have given me during my undergraduate and graduate studies.

I would also like to thank Professors Joseph Cheriyan, Ian Goulden, and Anil Maheshwari for the invaluable guidance and advice that they have given me throughout my studies.

Thank you, Sharat, for the great advice that you have given me since my first URA term, and for the pointers you gave me regarding minimum-norm optimization. Thanks Madison, for being an amazing friend, and for all of the help you have given me – be it practice talks, reading over written work, or even just listening to me vent. Thank you to the C&O lunch gang for being such a welcoming and supportive group; in particular, thank you Amena, Camryn, David A., Nathan, Mahtab, Paul, Rian, Ronen, and Santiago for your friendship. Thanks Anant, Roxanna, Ruilin, and Trista, for your friendship and for always being just a chat message away.

I am grateful to Carol and Melissa, for all of their help regarding the administrative logistics. I am especially thankful to Melissa, for giving me clear directions through the maze of thesis-completion deadlines and procedures.

Thank you, Sreepriya and Aparna, for being the best younger sisters one could ask for, and thanks, Sree, for frequently playing the role of counselor as well.

I am thankful to my aunts and uncles for their support; I am particularly grateful to my aunt Sathi, for her love and encouragement.

None of this would have been possible without the encouragement, love, and support given to me, and sacrifices made for me by my parents. I am grateful to my late father for his foresight, advice, and lessons that continue to guide me; and no words will ever be sufficient to thank my pillar of strength, my mother, who has played the role of both parents for me. I dedicate this thesis to them.

Dedication

To my parents

Table of Contents

List of Figures xi

List of Algorithms xii

1 Introduction 1

1.1 Our contributions 4

1.2 Related work 6

2 Preliminaries 10

2.1 Social choice theory 10

2.2 k-clustering 12

3 The single-winner problem ($k = 1$) 15

3.1 LP-duality based analysis framework 17

3.1.1 Copeland 22

3.1.2 Matching uncovered set 24

3.1.3 A sufficient condition for $\mathrm{OPT}(P^{\sigma}_{ao}) \leq 3$ 26

3.2 Randomized social choice functions 30

3.2.1 Computing an instance-optimal randomized SCF 31

3.2.2 A lower bound for the distortion of randomized SCFs 33
3.2.3 Extending the analysis framework to randomized SCFs 37

4 Multiwinner selection and the k-median problem **39**
4.1 Two k-median algorithms 41

5 k-median with limited value queries **44**
5.1 A blackbox reduction to the cardinal setting 45
5.2 Computing an estimate of OPT 48
5.3 Improving query complexity 51
5.4 Query complexity independent of n 55

6 Beyond social cost minimization: $O(1)$-distortion algorithms for the ℓ-centrum problem **58**
6.1 A blackbox reduction for the Top_ℓ setting 61
6.2 Improving query complexity 65
6.3 Adaptive sampling for ℓ-centrum k-clustering 72

7 Adaptive sampling for minimum-norm k-clustering **83**

8 Conclusions and Future Work **90**

References **92**

APPENDICES **97**

A Expansion of (Best-Dist) **98**

B Low distortion algorithms for other social cost minimization problems 100

B.1 Minimum spanning tree 101

B.2 Other social cost minimization problems 104

List of Figures

1.1 Definition of d in Example 1.1 3

3.1 Definition of d in Example 3.1 16
3.2 Network $\mathcal{N}_j$ corresponding to constraint (3.15) 20
3.3 Dual solution for $|ao| \geq |oa|$ 21
3.4 Dual solution for $j \in S$ 23
3.5 Dual solution for Theorem 3.1.4 25
3.6 Partial dual solution for $\{i, j, k_1, k_2\}$ 27
3.7 Partial dual solution for $\{i, j, k\}$ (if $|boa|$ is odd) 27
3.8 Metric $\tilde{d}$ when $2|oab| > |boa|$ 29
3.9 Metric $\tilde{d}$ when $2|oab| \leq |boa|$ 30
3.10 An optimal solution to the dual of (Best-Dist) 35

4.1 A k-winner selection instance with unbounded distortion 40

B.1 An MST instance with unbounded distortion 101

List of Algorithms

1 Online facility location algorithm with uniform costs [35] 41

2 Adaptive sampling algorithm for k-median [3] 43

3 A blackbox reduction to k-median 46

4 Minimum cost k-Forest via Boruvka's algorithm 50

5 $O(1)$-distortion, $O((\log(1/\delta) + \log k) \log n)$-query mechanism for k-median . 53

6 $O(1)$-distortion, $O(\log(1/\delta)k)$-query mechanism for k-median 56

7 A blackbox reduction to the ℓ-centrum problem 62

8 Extension of Meyerson's OFL algorithm for ℓ-centrum k-clustering 65

9 $O(1)$-distortion, $O((\log(1/\delta) + \log k) \log n)$-query mechanism for ℓ-centrum . 70

10 Adaptive sampling algorithm for ℓ-centrum 73

11 $O(1)$-distortion, $O(k \log(1/\delta) \log \ell)$-query mechanism for ℓ-centrum 81

12 Adaptive sampling algorithm for min-norm k-clustering (when t^* is known) 86

13 Adaptive sampling algorithm for minimum norm k-clustering 88

14 Procedure for finding undominated edges [9] 102

15 Algorithm for MST 103

16 Randomized 3-spanner algorithm [12] 104

Chapter 1

Introduction

In many applications, ranging from elections to network design problems, we wish to aggregate the preferences of agents in a given system and select an outcome that maximizes social welfare (i.e. the total value gained by the agents) or minimizes social cost (i.e. the total cost incurred by the agents). While we typically assume that agent preferences are captured by a *cardinal* utility function that assigns a numerical value to each outcome, in many contexts we only have access to *ordinal* information, namely the agents' preferences, or rankings over the outcomes. There are many reasons why such situations may arise; perhaps the most prominent is that the agents themselves may find it difficult to place numerical values on the possible outcomes [7]. It is also possible that, due to privacy and/or security concerns, the system designer is required to, or wishes to elicit less sensitive information. As ordinal preference rankings are not as expressive as cardinal utilities, a loss of efficiency in terms of the quality of the outcome computed is inevitable. Procaccia and Rosenschein [39] introduced the notion of *distortion* to quantify the worst-case efficiency loss for a given social choice function.

To describe this notion, and the underlying algorithm-design issues more meaningfully, we first consider the context of election problems, wherein each agent has a preference ordering over the set of candidates and we wish to elect a committee of k candidates that minimizes the *social cost* (i.e., the sum of costs incurred by the agents). More precisely, an instance of an election problem $(\mathcal{C}, A, \sigma)$ consists of a set of agents (or voters) $\mathcal{C}$, a set

of alternatives (or candidates) A, and a preference profile (a tuple giving the preference ordering for each agent), σ. The cost incurred by an agent i when candidate $a \in A$ is chosen is denoted as $d(i, a)$.

Following recent work, we will restrict our attention to metric social choice problems. In the metric setting, the agents and candidates are points in a metric space $(\mathcal{C} \cup A, d)$, where $d : (\mathcal{C} \cup A)^2 \to \mathbb{R}_{\geq 0}$ is a distance function satisfying the triangle inequality. This assumption models many applications, including those in which agents prefer alternatives that are ideologically similar to them, and hence, $d(i, a)$ can be intepreted as the *ideological distance* between agent i and alternative a. As the preference profile σ arises from the underlying cardinal costs induced by the metric, we will assume that d is *consistent* with σ; that is, for any $a, b \in A$, $i \in \mathcal{C}$, if i prefers a over b (denoted $a \succeq_i b$), then $d(a, i) \leq d(b, i)$. A social choice function maps a preference profile σ to a single alternative $f(\sigma) \in A$, i.e., specifies a solution to the 1-committee selection problem. The notion of distortion is akin to the notion of approximation ratio in the standard cardinal setting; for a given social choice function f, the distortion of f is the worst case ratio (over all instances) of the social cost of the candidate chosen by f over the minimum social cost. More formally, we define

$$\text{distortion}(f) = \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(f(\sigma), j)}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(o, j)}$$

where $d \triangleleft \sigma$ denotes that d is consistent with σ. Abusing notation slightly, given an instance $(\mathcal{C}, A, \sigma)$, we say that $a \in A$ is a ρ-distortion candidate with respect to σ if $\frac{\sum_{j \in \mathcal{C}} d(a,j)}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(o,j)} \leq \rho$. Note that if $f(\sigma)$ is always a ρ-distortion candidate for any preference profile σ, then f is a ρ-distortion social choice function.

The following example will illustrate this definition, and help us appreciate the difficulties that one faces when designing low-distortion social choice functions. One of the simplest and most natural social choice functions to consider is the plurality voting rule, wherein each agent casts one vote for their favourite candidate, and the candidate with the most number of votes is chosen. However, as shown by Example 1.1 (originally given by Anshelevich et. al [6]), the distortion of the plurality voting rule is $\Omega(m)$, where $m = |A|$ is the number of candidates.

Example 1.1. Let $m > 2$. The set of candidates is $A = \{a, b_1, \ldots, b_{m-1}\}$ and the set of agents (voters) is $\mathcal{C} = S_a \dot{\cup} S_{b_1} \dot{\cup} \cdots \dot{\cup} S_{b_{m-1}}$, where S_c is the set of agents whose top choice is

$c \in A$. For ease of notation, we will write $S_b = S_{b_1} \dot{\cup} \cdots \dot{\cup} S_{b_{m-1}}$. For some constant $q \geq 1$, $|S_a| = q + 1$ and $|S_{b_i}| = q$ for $i = 1, \ldots, m-1$. Note that the number of candidates is m and the number of voters is $mq + 1$. The preference rankings of the agents is:

- $j : a \succeq b_1 \succeq \cdots \succeq b_{m-1}$ for $j \in S_a$
- $j : b_i \succeq b_1 \succeq \cdots \succeq b_{i-1} \succeq b_{i+1} \succeq \cdots \succeq b_{m-1} \succeq a$ for $j \in S_{b_i}$, for $i = 1, \ldots, m-1$

According to the preference profile σ that is defined above, a is the unique winner according to the plurality voting rule. Consider the metric $(\mathcal{C} \cup A, d)$ defined by the following graph. The squares denote candidates and the dark circles indicate the location of all agents in the specified set. The distance between a pair of agents/candidates is the length of the shortest path in the graph. Notice that, in this graph, $b_1, \ldots, b_{m-1}$ are colocated (and hence $d(b_i, b_j) = 0$ for any $i, j \in \{1, \ldots, m-1\}$). The cost incurred by the agents when

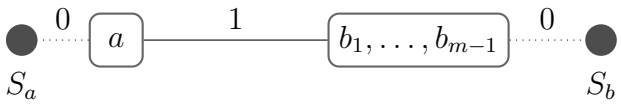


Figure 1.1: Definition of d in Example 1.1

a is chosen is $|S_a| \cdot 0 + |\mathcal{C} - S_a| \cdot 1 = (m-1) \cdot q$. The cost incurred by agents when b_i is chosen (for some $i \in \{1, \ldots, m-1\}$) is $|S_a| \cdot 1 + |\mathcal{C} - S_a| \cdot 0 = q + 1$. So,

$$\frac{\sum_{j \in \mathcal{C}} d(j, a)}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(j, o)} \geq \frac{(m-1)q}{q+1} \geq \frac{m-1}{2}$$

and thus a is an $\Omega(m)$-distortion candidate with respect to σ.

Another popular voting rule, which we will study in Chapter 3, is the Copeland voting rule. In this rule, the score of each candidate is the number of pairwise elections she has won; a candidate with the highest Copeland score wins the election. For the instance in Example 1.1, b_1 wins every pairwise election; that is, for any candidate $c \neq b_1$, at least half of the agents prefer b_1 to c. A candidate who wins all pairwise elections is a *Condorcet winner*, and it can easily be shown that if a is Condorcet winner, $\frac{\sum_{j \in \mathcal{C}} d(j,a)}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(j,o)} \leq 3$ for any d that is consistent with σ.

We sketch the proof of this fact, primarily to give some intuition as to how one could establish an upper bound on the quantity $\max_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(j,a)}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(j,o)}$. Suppose $a \in A$ is a Condorcet winner (for the instance $(\mathcal{C}, A, \sigma)$); consider any $o \neq a$ and d consistent with σ. We will show that $\sum_{j \in \mathcal{C}} d(j,a) \leq 3 \sum_{j \in \mathcal{C}} d(j,o)$. The cost incurred by any agent j who prefers a over o is easy to bound, as $d(j,a) \leq d(j,o)$. It remains to bound $d(j,a)$ for agents who prefer o over a; we do so using the triangle inequality. Since d satisfies the triangle inequality, $d(a,o) \leq d(a,j) + d(j,o)$ for any $j \in \mathcal{C}$ and hence if j prefers a over o, $d(a,o) \leq 2d(j,o)$. Since a is a Condorcet winner, at least $\frac{n}{2}$ agents prefer a over o, and thus $d(a,o) \leq \frac{\sum_{j \in \mathcal{C}: a \succeq_j o} 2d(j,o)}{\frac{n}{2}}$. Furthermore, $|\{j \in \mathcal{C} : o \succeq_j a\}| \leq \frac{n}{2}$, and hence $|\{j \in \mathcal{C} : o \succeq_j a\}| \cdot d(a,o) \leq \frac{n}{2} \cdot \frac{2}{n} \cdot \sum_{j \in \mathcal{C}: a \succeq_j o} 2d(j,o)$. Thus,

$$\sum_{j \in \mathcal{C}} d(j,a) \leq \sum_{j \in \mathcal{C}} d(j,o) + \sum_{j \in \mathcal{C}: o \succeq_j a} d(a,o) \leq \sum_{j \in \mathcal{C}} d(j,o) + \sum_{j \in \mathcal{C}: a \succeq_j o} 2d(j,o) \leq 3 \sum_{j \in \mathcal{C}} d(j,o)$$

In cases where the instance admits a Condorcet winner, (such as Example 1.1), the Condorcet winner is an obvious 3-distortion candidate. However, a well-known fact in social choice theory is that a Condorcet winner need not always exist. Furthermore, in order to bound the distortion of a social choice function f, we must bound $\max_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(j, f(\sigma))}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(j,o)}$ *for all* preference profiles σ (including those that do not admit a Condorcet winner). As it is difficult to reason about an arbitrary preference profile σ, bounding the distortion of a social choice function is a non-trivial task.

Given this sobering observation, it is perhaps surprising that $O(1)$-distortion (and in fact, 3-distortion) social choice functions exist [24] for the 1-median problem.

1.1 Our contributions

The main focus of this thesis is the problem of electing low-distortion k-committees. We begin, in Chapter 3, with the single-winner setting ($k = 1$) and give a novel LP-duality based analysis framework that makes it easier to analyze the distortion of existing social choice functions. Previously, Kempe [28] has also presented such a framework, however our framework is much simpler. Using this framework, we give simpler proofs of some known results.

While there is a tight bound of 3 on the distortion achievable by deterministic social choice functions [6, 24], obtaining tight bounds on the best distortion achievable by randomized SCFs is an open question. We do not know if randomized social choice functions can achieve asymptotically better distortion (as $|\mathcal{C}|, |A|$ grow) than deterministic SCFs, and a longstanding conjecture is that there exists a randomized SCF of distortion at most 2.

We formulate an LP to find an instance-wise optimal randomized social choice function; i.e., the LP computes, for a given instance, the randomized SCF with smallest distortion. (Note that this is fairly straightforward for deterministic SCFs, since the number of possible outputs for a given instance is a finite set, but for randomized SCFs, there is an infinite collection of distributions to search from.) Using this LP, we *disprove the above conjecture* by finding an instance for which the LP-optimum is larger than 2. We note that this conjecture has also been independently refuted by Charikar and Ramakrishnan [22], however our approach is quite different from theirs, and the instance-optimal LP we work with may be of independent interest.

For the k-committee selection problem (or equivalently, the k-median clustering problem) when $k > 1$, it is not possible to compute a $O(1)$-distortion committee using purely ordinal information [10]. We devise $O(1)$-distortion mechanisms for this k-median problem that use a limited number of value queries per agent, where a value query takes a pair of points i, j as input, and returns the distance $d(i, j)$ between them.

For simplicity, we consider the setting $A = \mathcal{C}$ here (as is often the case in clustering). Clearly, by making $n = |A| = |C|$ value queries per agent, or $O(n^2)$ queries in all, we can infer d precisely. We devise mechanisms with substantially better query complexity than this trivial bound that obtain $O(1)$-distortion. To the best of our knowledge, our results are the *first* results establishing upper bounds on distortion for k-median when $k > 1$.

- In Chapter 5, we devise a mechanism with $O(\log n \log k)$ queries per agent. This is based on a general reduction to the cardinal setting that is quite generic, and is of independent interest.

- A pertinent question that arises is whether one can obtain per-agent query complexity

that is independent of n. In Chapter 5, we answer this question affirmatively by devising a mechanism that makes $O(k)$ queries per agent.

Our mechanisms above are randomized mechanisms, that achieve $O(1)$ distortion with constant success probability.

We next consider more general k-clustering problems, where the objective to be minimized is a *monotone, symmetric norm* of the agents' costs (a norm f is monotone, if $f(x) \leq f(y)$ whenever $0 \leq x \leq y$, and symmetric if $f(v) = f(\pi(v))$ for any permutation π). In Chapter 6, we consider the ℓ-centrum problem, wherein we seek to minimize the sum of the ℓ largest costs. (The underlying norm here is called the Top_ℓ norm: $\text{Top}_\ell(v)$ is the sum of the ℓ largest entries of v in absolute value.) We adapt our k-median mechanisms to obtain analogous results for ℓ-centrum: we obtain O(1)-distortion mechanisms with per-agent query complexities of $O(\log k \log n)$ (see Theorem 6.2.4) and $O(k \log \ell)$ (see Theorem 6.3.13). The latter result is obtained via a simple and elegant adaptive-sampling algorithm.

Finally, in Chapter 7, we consider the completely general setting where the objective is an *arbitrary* monotone, symmetric norm, a problem called *minimum-norm k-clustering*, and show how to leverage the adaptive-sampling idea to obtain a constant-factor bicriteria approximation algorithm for this problem (see Theorem 7.0.10).

The primary focus of this thesis is the k-committee selection problem; however, we note that it is also possible to obtain low query-complexity, $O(1)$-distortion mechanisms for other social cost minimization problems. We defer these results to Appendix B.

1.2 Related work

The distortion of social choice functions was first studied by Procaccia and Rosenschein [39]. In this thesis, we restrict ourselves to the metric social choice setting, where we assume that the agents and candidates are points in a metric space $(\mathcal{C} \cup A, d)$. The distortion of social choice functions in the metric setting was first studied by Anshelevich, Bharadwaj, and Postl [6], who proved that the Copeland voting rule has a distortion of 5. They also

showed that every deterministic social choice function has a distortion of at least 3, and conjectured that there exists a deterministic social choice function with distortion of at most 3. The *matching uncovered set* is an elegant construction that arose during the quest for a 3-distortion social choice function. It was first introduced by Munagala and Wang [37], who proved that any candidate in the matching uncovered set is a 3-distortion candidate (Kempe [28] also showed this using an LP-duality based framework). The conjecture was ultimately resolved by Gkatzelis, Halpern, and Shah [24], who gave a 3-distortion social choice function and in doing so, also proved that the matching uncovered set is not empty. More recently, Kizilkaya and Kempe gave a much simpler 3-distortion social choice function [30].

The distortion of randomized social choice functions in the metric setting has also been studied. Notably, Anshelevich and Postl [8] gave a simple randomized social choice function that has a distortion of $3 - \frac{2}{|\mathcal{C}|}$, and Kempe and Gkatzelis et. al gave randomized social choice functions that achieve a distortion of $3 - \frac{2}{|A|}$ [29, 24]. A longstanding conjecture was that there exists a randomized social choice function that achieves a distortion of 2; as noted earlier, this conjecture was refuted independently by our work [40] and Charikar and Ramakrishnan [22].

The work most closely related to ours is Caragiannis, Shah, and Voudouris [18], who also study a metric multiwinner election problem wherein one seeks to elect a committee of k winners so as to minimize the sum of costs incurred by the agents. They consider the setting where the cost of an agent for a committee is her distance from the q-th closest alternative in the committee. They show that the distortion of any social choice correspondence is unbounded when $q \leq \frac{k}{3}$, but one can give a $\Theta(n)$-distortion voting rule when $q \in (\frac{k}{3}, \frac{k}{2}]$, and a $O(1)$-distortion voting rule when $q > \frac{k}{2}$. Our approach is different in that we only consider the setting where $q = 1$; however, we circumvent the impossibility result by using a limited amount of cardinal information. The metric multiwinner election problem has also been studied by Goel, Hulett, and Krishnaswamy [25] and Chen, Li, and Wang [23], however these models are quite different from ours; Chen et. al assume that the locations of the candidates are public knowledge, and Goel et. al's model assumes that the cost of an agent for a committee is the sum of her distances from *every* member of the committee.

There have been many other interesting directions pursued in the metric distortion literature. Anagnostides, Fotakis, and Patsilinakos [5] studied the distortion of metric social choice functions when incomplete ordinal information is provided. They also studied the problems where we are given partial rankings over the candidates, and where we do not have the preference ranking of all voters. Goel, Krishnaswamy, and Munagala [26] studied the distortion of the Copeland voting rule and other social choice functions under the $\max_{\ell \in [n]} \mathrm{Top}_\ell$ objective (which they call the *fairness ratio*).

So far, we have only discussed work done in the social-cost-minimization setting. A significant amount of work has also been done in the social-welfare-maximization setting. In the social welfare maximization setting (where we do not have the metric assumption), Caragiannis and Procaccia [17] showed that the Plurality rule has a distortion of $O(m^2)$, which was later proven to be the best possible among all *deterministic* voting rules by Caragiannis, Filos-Ratsikas, Nath, and Voudouris [16]. Borodin, Halpern, Latifian, and Shah [13] studied the problem of selecting a committee of size k when given the top-t preferences of the voters over the candidates. There has also been work done pertaining to other optimization problems. Notably, Anshelevich and Sekar [9] combine randomization with an intuitive greedy algorithm to design a 1.6-distortion algorithm for maximum weight matching; using similar techniques, Abramowitz and Anshelevich [1] gave $O(1)$-distortion algorithms for a sizeable class of graph optimization problems (including maximum weight spanning tree, maximum b-matchings, etc.).

In light of these results, a natural question to ask is whether eliciting a small amount of additional cardinal information from the agents can yield better algorithms. Amanatidis, Birmpas, Filos-Ratsikas, and Voudouris [4] studied mechanisms for single winner elections in the social-welfare maximization setting that elict a few value or comparison queries per agent. In the social cost minimization setting, Abramowitz, Anshelevich, and Zhu [2] studied metric single winner elections when some aggregated cardinal information is available. Anshelevich and Zhu studied some graph optimization problems in the setting where the locations of the candidates are known, but the locations of the voters is private information [10]. However, to the best of our knowledge, little is known regarding the performance of mechanisms that elicit (a few) queries per agent in the social cost minimization setting.

The committee selection problem we study in this thesis has strong connections to

k-clustering, and many of the mechanisms we design will use clustering algorithms as subroutines. Moreover, in Chapters 6 and 7 we present adaptive sampling algorithms for the ℓ-centrum and minimum-norm k-clustering problems. In the ℓ-centrum k-clustering problem, one seeks to open k centers so as to minimize the sum of the ℓ-largest assignment costs. Thus, this problem captures both the k-median clustering problem (wherein one seeks to minimze the sum of all assignment costs) and the k-center clustering problem (where one wishes to minimize the maximum assignment cost). The ℓ-centrum and its generalization, the ordered k-median problem, have been extensively studied in the Operations Research literature (see, e.g., [11, 32, 38] and the references within). The first $O(\log n)$-approximation algorithms for ℓ-centrum and the ordered k-median problems were given by Tamir [42] and Aouad and Segev [11] respectively. Within the past few years, Byrka, Sornat, and Spoerhase [15], and Chakrabarty and Swamy [19] gave the first constant-factor approximations for these two problems. More recently, Chakrabarty and Swamy [20] gave the first constant factor approximation algorithm for the minimum-norm k-clustering problem. Using LP-techniques, they devised a $(408 + \varepsilon)$-approximation algorithm that opens exactly k centers. Note that the minimum-norm k-clustering problem generalizes many well-studied k-clustering problems, including ℓ-centrum and ordered k-median.

Chapter 2

Preliminaries

2.1 Social choice theory

Social choice theory is concerned with aggregating the preferences of multiple individuals or agents into a single outcome. Let $\mathcal{C}$ be a set of n agents, and let A be a finite set of alternatives. For $a_1, a_2 \in A$, and $i \in \mathcal{C}$, we say that $a_1 \succeq_i a_2$ if agent i prefers alternative a_1 to a_2. Agent i's preference relation induces a preference ordering $\succeq_i$, which is a total ordering on A. We denote the top choice of $i \in \mathcal{C}$ as $top(i, \succeq_i)$, or just $top(i)$ when $\succeq_i$ is clear from the context. Similarly, We denote the top choice of $i \in \mathcal{C}$ when restricted to $S \subseteq A$ as $top_S(i, \succeq_i)$, or just $top_S(i)$ when $\succeq_i$ is clear from the context. Let L be the set of total orders on A. A preference profile $\sigma = (\succeq_1, \ldots, \succeq_n) \in L$ is a tuple giving a preference ordering for each agent.

As alluded to earlier, we are interested in settings where the ordinal preferences σ are a result of the underlying cardinal utility functions of the agents. To be more precise, we will assume that the agents and alternatives are points in a metric space $(\mathcal{C} \cup A, d)$, where $d : (\mathcal{C} \cup A)^2 \to \mathbb{R}_{\geq 0}$ is a distance function satisfying the triangle inequality. The cost incurred by an agent i when alternative $a \in A$ is chosen is the distance between herself and a, which we will denote as $d(i, a)$. As the preference profile σ arises from the underlying cardinal costs induced by the metric, we will assume that d is *consistent* with σ; that is, for any $a, b \in A$, $i \in \mathcal{C}$, if $a \succeq_i b$, then $d(a, i) \leq d(b, i)$.

Definition 2.1.1 (Social Choice Functions and Correspondences [41])**.** A *social choice function* (SCF) $f : L \to A$ is a function which maps a preference profile σ to a single alternative in A. A *social choice correspondence* (SCC) $f : L \to 2^A$ is a function which maps a preference profile σ to a subset of alternatives in A [1].

We introduce a piece of non-standard terminology and say that f is a *social choice k-correspondence*, or simply *k-correspondence*, if $f : L \to A^k$, i.e., f maps preference profiles to subsets of A of size k. Notice that every social choice k-correspondence is a social choice correspondence, and a social choice 1-correspondence is in fact a social choice function. Thus, the problem(s) described in Chapter 1 can be regarded as the problem of finding a good social choice k-correspondence. As ordinal preference rankings are not as expressive as cardinal utilities, a loss of efficiency, in terms of the quality of the outcome computed, is inevitable. Procaccia and Rosenschein [39] introduced the notion of *distortion* to quantify the worst-case efficiency loss for a given social choice function. This notion extends readily to social choice correspondences.

Definition 2.1.2 (Distortion of a Social Choice k-Correspondence)**.** Let $f : L \to A^k$ be a social choice k-correspondence. We define the distortion of f to be

$$\text{distortion}(f) = \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(j, f(\sigma))}{\min_{S \subseteq A^k} \sum_{j \in \mathcal{C}} d(j, S)}$$

where $d \triangleleft \sigma$ denotes that d is consistent with σ.

As we will see in Chapter 4, for $k \geq 3$, any social choice k-correspondence has unbounded distortion, so algorithms need to use some additional cardinal information to achieve bounded distortion. We will therefore work with *mechanisms*, which are algorithms that can make additional queries. The following definition of a mechanism is a slight modification of the definition given by Amanatidis, Birmpas, Filos-Ratsikas, and Voudouris [4].

[1] It is natural to ask whether a social choice correspondence can simply be viewed as an SCF with an expanded alternative-set 2^A. This is not quite true because even though in many settings, an ordering over A can be used to naturally induce an ordering over 2^A, this does not capture the space of all possible orderings of 2^A

Definition 2.1.3 (Mechanism [4])**.** A *mechanism* $\mathcal{M} = (\mathcal{Q}, f, k)$ with access to a query oracle takes as input a preference profile σ and returns a k-subset of A. It consists of an algorithm $\mathcal{Q}$ that, given an input preference profile σ, adaptively makes queries to the query oracle, and a function f that takes the input σ, the set of queries and their answers, and outputs a k-subset of A (i.e., a solution).

Since the output of $\mathcal{M}$ depends on the underlying distance function d we will often write $\mathcal{M}(\sigma|d)$ to denote the output of $\mathcal{M}$ given σ and a metric d that is consistent with σ. To be clear, the mechanism is *not* provided d as input; however, the answers provided by the querying algorithm $\mathcal{Q}$, and hence the final output of the mechanism does depend on d. We also extend the notion of distortion to mechanisms in the natural manner.

Definition 2.1.4 (Distortion of a Mechanism)**.** Let $\mathcal{M} = (\mathcal{Q}, f, k)$ be a mechanism. We define the distortion of $\mathcal{M}$ to be

$$\text{distortion}(\mathcal{M}) = \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(j, \mathcal{M}(\sigma|d))}{\min_{S \subseteq A^k} \sum_{j \in \mathcal{C}} d(j, S)}$$

where $d \triangleleft \sigma$ denotes that d is consistent with σ.

Finally, we note that many of the mechanisms we design in this thesis will be randomized. For a randomized mechanism, wherein the solution returned is a random variable (or equivalently, where the output is a distribution over solutions), the most natural notion of distortion is the worst-case ratio between the *expected* cost of the solution returned and the optimum. However, in Chapters 5 and 6, we will use the somewhat weaker notion, where we say that a randomized mechanism has distortion of at most ρ if $\sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(j, \mathcal{M}(\sigma|d))}{\min_{S \subseteq A^k} \sum_{j \in \mathcal{C}} d(j, S)} \leq \rho$ with constant probability.

2.2 k-clustering

When designing low-distortion mechanisms for the committee selection problem, we will often use k-clustering algorithms as subroutines of our mechanisms. We introduce the pertinent clustering problems here.

For a vector v, we use $v^{\downarrow}$ to denote v with its coordinates sorted in non-increasing order. For $v \geq 0$, the Top-ℓ cost of v, denoted $\text{Top}_\ell(v)$, is the sum of the ℓ largest entries of v. Equivalently, $\text{Top}_\ell(v) = \sum_{i=1}^{\ell} v_i^{\downarrow}$. A norm $f : \mathbb{R}^n \to \mathbb{R}$ satisfies (i) $f(x) = 0$ iff $x = 0$; (ii) $f(x + y) \leq f(x) + f(y)$ for all $x, y \in \mathbb{R}^n$ (triangle inequality); and (iii) $f(\lambda x) = |\lambda| f(x)$ for all $x \in \mathbb{R}^n$, $\lambda \in \mathbb{R}$ (homogeneity). We say that f is *symmetric* if $f(v) = f(v^{\downarrow})$ for all $v \in \mathbb{R}^n$. A *monotone* norm f satisfies $f(x) \leq f(y)$ whenever $0 \leq x \leq y$.

In the k-clustering problem, we have an integer $k \geq 0$ and a metric space $(\mathcal{C}, d)$, where $\mathcal{C}$ is a set of n agents and $d(i, j)$ is the distance between two agents i and j. We wish to open a set of centers $S \subseteq \mathcal{C}$, and assign each agent to its nearest open center in S. The distance between an agent j and its nearest open center in S, denoted $d(j, S)$, is the assignment/connection cost incurred by j under the current solution. Thus, a set of open centers S induces an assignment cost vector, denoted $d(\mathcal{C}, S)$, where $d(\mathcal{C}, S)_j = d(j, S)$ is the assignment cost incurred by agent j under S. Two classical k-clustering problems are the *k-center* and *k-median* problems. In the k-center problem, we wish to minimize the largest assignment cost, whereas in the k-median problem, we wish to minimize the sum of all assignment costs. A generalization of these two clustering problems is the *ℓ-centrum k-clustering problem* (or simply ℓ-centrum), in which we wish to open k centers so as to minimize the sum of the ℓ largest assignment costs. In *minimum-norm k clustering*, we wish to minimize $f(d(\mathcal{C}, S))$, where f is a monotone symmetric norm, and $d(\mathcal{C}, S)$ is the assignment cost vector induced by S. Notice that Top_ℓ is a monotone symmetric norm, so ℓ-centrum is a special case of minimum-norm k-clustering.

In general, the Top_ℓ objective can be difficult to work with, due to its non-separable nature: the contribution of an agent depends on the assignment costs incurred by the other agents. In order to design a good algorithm for the ℓ-centrum problem, we must first overcome this inherent dependence on the relative ordering of the agents with respect to assignment cost. As a means to this end, we will consider the proxy function introduced by Chakrabarty and Swamy [20]. The following basic result will be quite useful.

Claim 2.2.1 (Claims 6.1 and 6.2 in [20]). *For any $v \in \mathbb{R}^n_+$, $\rho \in \mathbb{R}$, $\text{Top}_\ell(v) \leq \ell \cdot \rho + \sum_{j=1}^{n} (v_j - \rho)^+$. Moreover, if $v_\ell^{\downarrow} \leq \rho \leq (1+\varepsilon) v_\ell^{\downarrow}$, then $\ell \cdot \rho + \sum_{j=1}^{n} (v_j - \rho)^+ \leq (1+\varepsilon) \cdot \text{Top}_\ell(v)$.*

Claim 2.2.1 shows that for a suitable choice of ρ, the function $\ell \cdot \rho + \sum_{j=1}^{n} (v_j - \rho)^+$

serves as a good proxy for $\mathrm{Top}_\ell(v)$. The key benefit of this proxy function is that for a fixed ρ, the $\ell \cdot \rho$ term is a constant, and so we can work with the expression $\sum_j (v_j - \rho)^+$, which *is* separable across the coordinates.

Chapter 3

The single-winner problem ($k = 1$)

In the single winner election problem, we are given a set of agents $\mathcal{C}$ and a set of candidates (or alternatives) A that are located in a metric space $(\mathcal{C} \cup A, d)$, where $d(i, a)$ is the distance from agent i to candidate a. We would like to choose a candidate $a \in A$ that minimizes the social cost (i.e. $\sum_{j \in \mathcal{C}} d(j, a)$). As a clustering problem, this is precisely the 1-median problem. Note however that we only know σ, the preference profile of the agents over the candidates. Recall that we define the distortion of a social choice function (SCF) f to be $distortion(f) = \sup_\sigma \sup_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(j, f(\sigma))}{\min_{a \in A} \sum_{j \in \mathcal{C}} d(j, a)}$.

A longstanding question was whether there exists a social choice function with distortion *at most* 3. This was recently resolved by Gkatzelis, Halpern, and Shah [24], who defined a social choice function PluralityMatching and proved that it has distortion at most 3. In fact, this is the best possible bound; as shown by the following example, originally given by Anshelevich, Bhardwaj, Elkind, Postl, and Skowron [6], the distortion of any social choice function must be at least 3.

Example 3.1. Consider a social choice function f, and an instance $(\mathcal{C}, A, \sigma)$, where $A = \{x, y\}$. Let $n = \frac{|\mathcal{C}|}{2}$. Suppose $\frac{n}{2}$ of the agents prefer x over y (the other $\frac{n}{2}$ agents prefer y over x) and assume, without loss of generality, that $f(\sigma) = x$. Consider the metric $(\mathcal{C} \cup A, d)$ defined by the following graph. The squares denote candidates and the dark circles indicate the location of all agents in the specified set ($xy = \{j \in \mathcal{C} : x \succeq_j y\}$,

$yx = \{j \in \mathcal{C} : y \succeq_j x\}$). The distance between a pair of agents/candidates is the length of the shortest path in the graph. Notice that d is consistent with the preference profile σ.

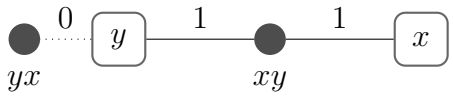


Figure 3.1: Definition of d in Example 3.1

The social cost when x is chosen is $\frac{n}{2} + \frac{n}{2} \cdot 2 = \frac{3n}{2}$. On the other hand, the social cost when y is chosen is $\frac{n}{2}$. Thus, $\frac{\sum_{j \in \mathcal{C}} d(j,x)}{\sum_{j \in \mathcal{C}} d(j,y)} = 3$. Since the assumption that $f(\sigma) = x$ was without loss of generality, and this is a metric that is consistent with σ, the distortion of f is at least 3.

In general, proving an upper bound on the distortion of a social choice function f can be a strenuous task; hence, analysis frameworks for proving bounds on the distortion of SCFs are valuable. Kempe gave such a framework [28], based on LP duality with the intent of simplifying and unifying certain proofs, but even this framework is somewhat involved. We present an LP-duality based analysis framework that is much simpler than Kempe's framework. In the deterministic setting, we show that it is possible to give simpler proofs of known distortion bounds using our framework. We also show that this framework can be leveraged to derive a simple sufficient condition for 3-distortion candidacy (see Lemma 3.1.5). Our framework has the added benefit that it generalizes readily to the randomized setting. Finally, we formulate an LP that computes, for any given instance, a randomized social choice function with optimal distortion (see (Best-Dist)). Using this, we obtain an instance for which the minimum achievable distortion is at least 2.063164, thereby disproving a widely-believed conjecture that there exists a *randomized* social choice function of distortion at most 2. (We note that independently and concurrently, Charikar and Ramakrishnan [22] have also disproved this conjecture: they obtain a slightly better bound of 2.1126, but their techniques are quite different; in particular, they do not obtain an instance-optimal randomized SCF).

3.1 LP-duality based analysis framework

In order to bound the distortion of a social choice function f, we wish to be able to compute the value of $\max_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(a,j)}{\sum_{j \in \mathcal{C}} d(o,j)}$ for any $a, o \in A$. This quantity can be computed by solving the following linear program. In all linear programs presented in this section, we will define variables d_{ij} which represent the distance $d(i,j)$, for any $i, j \in \mathcal{C} \cup A$. Let $m = |A|$ be the number of candidates and $n = |\mathcal{C}|$ be the number of agents, and let $\text{alt}_\sigma(k,r)$ denote the rth ranked outcome in agent k's ordering, $\succeq_k \in \sigma$.

$$\max \quad \sum_{j \in \mathcal{C}} d_{aj} \tag{Q^σ_{ao}}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{C}} d_{oj} \leq 1 \tag{3.1}$$

$$d_{\text{alt}_\sigma(j,r),j} \leq d_{\text{alt}_\sigma(j,r+1),j} \qquad \forall j \in \mathcal{C},\ \forall r \in [m-1] \tag{3.2}$$

$$d_{i_1 i_2} \leq d_{i_1 i_3} + d_{i_2 i_3} \qquad \forall i_1, i_2, i_3 \in A \tag{3.3}$$

$$d_{j_1 j_2} \leq d_{j_1 j_3} + d_{j_2 j_3} \qquad \forall j_1, j_2, j_3 \in C \tag{3.4}$$

$$d_{j_1 j_2} \leq d_{i_1 j_1} + d_{i_1 j_2} \qquad \forall i_1 \in A,\ j_1, j_2 \in C \tag{3.5}$$

$$d_{i_1 j_1} \leq d_{i_1 j_2} + d_{j_1 j_2} \qquad \forall i_1 \in A,\ j_1, j_2 \in C \tag{3.6}$$

$$d_{i_2 j} \leq d_{i_1 j} + d_{i_1 i_2} \qquad \forall i_1, i_2 \in A,\ j \in C \tag{3.7}$$

$$d_{i_1 i_2} \leq d_{i_1 j} + d_{i_2 j} \qquad \forall i_1, i_2 \in A,\ j \in C \tag{3.8}$$

$$d \geq 0 \tag{3.9}$$

Constraint (3.1) normalizes $\sum_{j \in \mathcal{C}} d_{oj}$, which allows us to avoid writing a ratio in the objective. Constraint (3.2) ensures that the metric d is consistent with the preference profile σ. Constraints (3.3)-(3.9) enforce that d is a metric (i.e., it satisfies the triangle inequality and non-negativity).

In order to prove that distortion$(f) \leq \rho$, it would suffice to show that, for all σ and for all $o \in A$, $(Q^\sigma_{f(\sigma),o})$ has an optimal value of at most ρ (this is equivalent to $\max_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(j,a)}{\sum_{j \in \mathcal{C}} d(j,o)} \leq \rho$ for all $o \in A$). One way to do this would be to demonstrate a dual solution of value at most

ρ. Unfortunately, the dual of (Q^{σ}_{ao}) can be difficult to interpret. Kempe's LP-duality based framework [28] addresses this challenge by considering a restricted set of dual solutions, which can be interpreted as flows in a particular network. Alternatively, we can work with a relaxation of (Q^{σ}_{ao}); the dual of this relaxation will have fewer variables, and hence hopefully be of a simpler form and easier to interpret.

To construct an appropriate relaxation of (Q^{σ}_{ao}), we note that the constraints (3.3)-(3.8) describe four types of 3-sets or "triangles": triangles consisting of three agents (3.4), two agents and a single candidate (3.5)-(3.6), one agent and two candidates (3.7)-(3.8), and triangles consisting of three candidates (3.3). (Note that given (3.2), constraint (3.7) is non-trivial when $i_1 \succeq_j i_2$). We can obtain a relaxation of (Q^{σ}_{ao}) with fewer constraints by only enforcing the triangle inequality for triangles consisting of one agent and two candidates, and furthermore, where one of the candidates is o. The motivation behind this relaxation is that various proofs can be viewed in terms of leveraging the triangle inequality with only these types of triangles.

More formally, define $\mathcal{T}(\sigma, o) := \{(j, i_1, i_2) : i_1, i_2 \in A : i_1 \neq i_2, i_1 \succeq_j i_2, o \in \{i_1, i_2\}, j \in \mathcal{C}\}$. We obtain the following LP from (Q^{σ}_{ao}) by dropping all constraints (3.3)-(3.6), as well as the constraints (3.7)-(3.8) that correspond to triangles that are not in $\mathcal{T}(\sigma, o)$.

$$\max \quad \sum_{j \in \mathcal{C}} d_{aj} \tag{P^{σ}_{ao}}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{C}} d_{oj} \leq 1 \tag{3.10}$$

$$d_{\text{alt}_\sigma(j,r),j} \leq d_{\text{alt}_\sigma(j,r+1),j} \qquad \forall j \in \mathcal{C},\ \forall r \in [m-1] \tag{3.11}$$

$$d_{i_2 j} \leq d_{i_1 j} + d_{i_1 i_2} \qquad \forall (j, i_1, i_2) \in \mathcal{T}(\sigma, o) \tag{3.12}$$

$$d_{i_1 i_2} \leq d_{i_1 j} + d_{i_2 j} \qquad \forall (j, i_1, i_2) \in \mathcal{T}(\sigma, o) \tag{3.13}$$

$$d \geq 0 \tag{3.14}$$

As (P^{σ}_{ao}) is a relaxation of (Q^{σ}_{ao}), any upper bound for OPT(P^{σ}_{ao}) is also an upper bound for OPT(Q^{σ}_{ao}). We will use (D^{σ}_{ao}), the dual of (P^{σ}_{ao}), to demonstrate upper bounds for OPT(P^{σ}_{ao}). For ease of notation, let $\mathcal{T} = \mathcal{T}(\sigma, o)$. The indicator variable $\mathbb{I}_{[i=\text{alt}_\sigma(k,r)]} = 1$ if

i is the rth ranked outcome in $\succeq_k$, and 0 otherwise. The dual of (P^σ_{ao}) is given below. For notational convenience, we define $\beta_{0,k} = \beta_{n,k} = 0$ (so they are not variables in the following LP).

$$\min \quad \gamma \tag{D^σ_{ao}}$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{T=(k,\cdot,i)\in\mathcal{T}} (\alpha_{1,T} - \alpha_{2,T}) - \sum_{T=(k,i,\cdot)\in\mathcal{T}} (\alpha_{1,T} + \alpha_{2,T}) + \beta_{1k}\mathbb{I}_{[i=\text{alt}_\sigma(k,1)]} \\
& \quad + \sum_{r=1}^{n} (\beta_{r+1,k} - \beta_{r,k})\mathbb{I}_{[i=\text{alt}_\sigma(k,r+1)]} - \beta_{n-1,k}\mathbb{I}_{[i=\text{alt}_\sigma(k,n)]} \\
& \quad + \gamma\mathbb{I}_{[i=o]} \geq \mathbb{I}_{[i=a]} \qquad \forall k \in \mathcal{C},\ \forall i \in A \qquad (3.15) \\
& \sum_{T\in\mathcal{T}: T=(\cdot,i_1,i_2) \text{ or } (\cdot,i_2,i_1)} (\alpha_{2,T} - \alpha_{1,T}) \geq 0 \qquad \forall i_1, i_2 \in A \qquad (3.16) \\
& \alpha, \beta, \gamma \geq 0 \qquad (3.17)
\end{aligned}$$

In (D^σ_{ao}), γ corresponds to the primal constraint (3.10), the β-variables correspond to the primal constraints (3.11), and the $\alpha_{1,\cdot}$ and $\alpha_{2,\cdot}$ variables correspond to the primal constraints (3.12) and (3.13) respectively.

It will be helpful to interpret a feasible solution to (D^σ_{ao}) as a flow in the following network. For every $j \in \mathcal{C}$, consider a flow in $\mathcal{N}_j$ (Figure 3.2) from the source o to the sink a, which has a demand, i.e., net in-flow requirement, of 1. The $-\alpha_{2,(j,i_1,i_2)}$ variables can be interpreted as additional charges on i_1 and i_2; constraint (3.15) requires that the net in-flow for node a must be at least 1, and for all other nodes, the net in-flow must be at least 0.

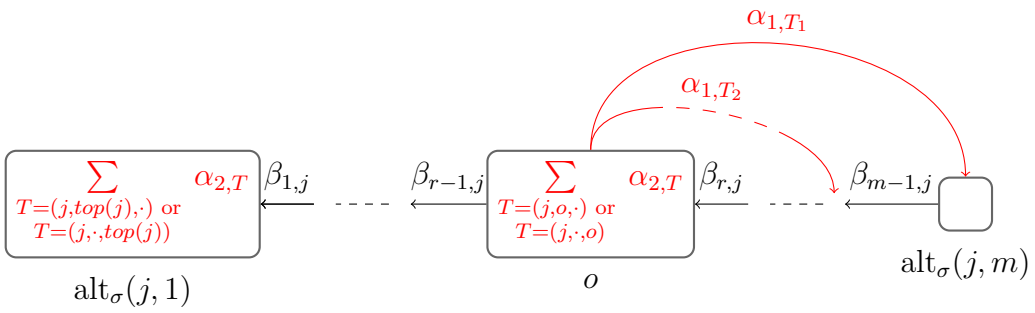
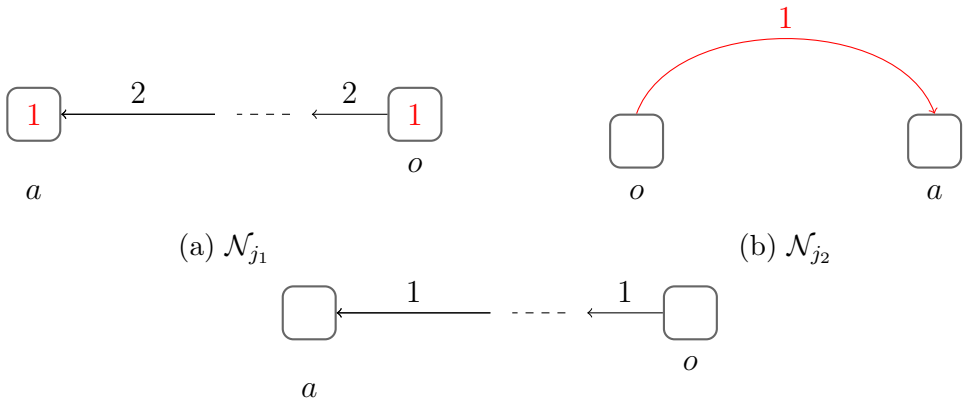


Figure 3.2: Network $\mathcal{N}_j$ corresponding to constraint (3.15)

When routing flow from o to a in $\mathcal{N}_j$, the black $\beta_{\cdot j}$ arcs can be used without incurring any additional cost. Sending flow via the red $\alpha_{1,T}$-arcs does incur a cost; by constraint (3.16), the amount of flow sent along any $\alpha_{1,(j,i_1,i_2)}$-arc must be charged to $\alpha_{2,T'}$ for some $T' = (\cdot, i_1, i_2)$ or $(\cdot, i_2, i_1) \in \mathcal{T}$. This additional $\sum_{T=(j,i,\cdot)} \alpha_{2,T}$ charge on an agent j is indicated in Figure 3.2 as a red number inside the square corresponding to candidate i. Finally, γ is the sum of the total flow departing from o and the $\alpha_{2,T}$-charges on o.

To be more concrete, consider the following instance $(\mathcal{C}, A, \sigma)$ with $\mathcal{C} = \{j_1, j_2, j_3\}$ and $A = \{a, o\}$. In this example and in the sequel, for any $x, y, z \in A$, we use xy to denote the set of agents who prefer x to y (i.e. $xy = \{j \in \mathcal{C} : x \succeq_j y\}$) and $xyz = \{j \in \mathcal{C} : x \succeq_j y \succeq_j z\}$. The preference profile σ is $ao = \{j_1, j_3\}$ and $oa = \{j_2\}$. That is, j_1 and j_3 prefer a to o, while j_2 prefers o to a. We can construct the following feasible dual solution, which has $\gamma = 3$ (proving that $\text{OPT}(D^\sigma_{ao}) \leq 3$ for this instance).



(a) $\mathcal{N}_{j_1}$ (b) $\mathcal{N}_{j_2}$

(c) $\mathcal{N}_{j_3}$

Figure 3.3: Dual solution for $|ao| \geq |oa|$

As depicted in Figure 3.3, in $\mathcal{N}_{j_3}$, one unit of flow is sent from o to a using $\beta_{\cdot,j_3}$ arcs. In $\mathcal{N}_{j_2}$, one unit of flow is sent from o to a using an $\alpha_{1,(j_2,o,a)}$ arc. The cost of this usage is charged to j_1, and hence we have $\alpha_{2,(j_1,a,o)} = 1$ (this is denoted as the red 1 inside the a and o nodes in Figure 3.3a). Finally, in $\mathcal{N}_{j_1}$, we route 2 units of flow from o to a along $\beta_{\cdot,j_1}$ arcs (1 unit of flow is to satisfy the demand of a, and the other unit is to neutralize the charge of 1 on a). To compensate for the total amount of flow leaving o, as well as the additional charge of 1 on o, we must have $\gamma = 3$.

This example can be generalized to prove that, for any instance with $|ao| \geq |oa|$, $\text{OPT}(D^{\sigma}_{ao}) \leq 3$. Since $|ao| \geq |oa|$, we can match every $j \in oa$ with a unique $j' \in ao$. The flow for j is as depicted in Figure 3.3b; the cost incurred by using the $\alpha_{1,(j,o,a)}$-arc is charged to j', whose flow is as depicted in Figure 3.3a. Finally, the flow for agents in ao who have not been matched to an agent in oa is as depicted in Figure 3.3c.

Fact 3.1.1. If $|ao| \geq |oa|$, then $\text{OPT}(P^{\sigma}_{ao}) \leq 3$

In the rest of this section, we will demonstrate how this framework can be used to give simpler proofs of some known results.

3.1.1 Copeland

A popular voting rule is the Copeland rule. In this rule, the score of each candidate is the number of pairwise elections she won; a candidate with the highest Copeland score is returned. Anshelevich, Bhardwaj, and Postl [6] proved that the distortion of the Copeland voting rule is 5. We use our analysis framework to give a simpler proof that the distortion of Copeland's rule is at most 5.

Theorem 3.1.2. *Let $(\mathcal{C}, A, \sigma)$ be an instance of the single winner election problem and let a be a Copeland winner. Then $OPT(D^{\sigma}_{ao}) \leq 5$ for all $o \in A \setminus \{a\}$.*

Proof. Recall that for $x, y, z \in A$, $xy = \{j \in \mathcal{C} : x \succeq_j y\}$ and $xyz = \{j \in \mathcal{C} : x \succeq_j y \succeq_j z\}$. Let $a \in A$ be a Copeland winner, and let $o \in A \setminus \{a\}$. If $|ao| \geq |oa|$, then by Fact 3.1.1, $\mathrm{OPT}(D^{\sigma}_{ao}) \leq 3$. Otherwise, if $|ao| < |oa|$, there exists $b \in A \setminus \{a, o\}$ such that $|ab| \geq |ba|$ and $|bo| \geq |ob|$ [36].

We now utilize b to construct a dual solution of value at most 5. We first describe the construction at a high level. We define a set S such that (1) $|ao \setminus S| \geq |oa \setminus S|$ and (2) we can partition S into pairs (i, j) such that the $\alpha_{1,T}$-arcs used in $\mathcal{N}_i$ are charged to nodes in $\mathcal{N}_j$ and vice versa. As $|ao \setminus S| \geq |oa \setminus S|$, for $j \notin S$, we can use the same construction as shown in Figure 3.3. Then, if we can show that the total cost on the source node in $\mathcal{N}_j$ is at most 5 for all $j \in S$, our dual solution would have value $\gamma \leq 5$.

Claim 3.1.3. *There always exists a set $S \subseteq oab \cup boa$ such that $|S \cap oab| = |S \cap boa|$ and $|ao \setminus S| \geq |oa \setminus S|$.*

Proof. First, consider the case where $|oab| \leq |boa|$ define S to be a set containing all agents in oab and $|oab|$ agents from boa. Then, $|ao \setminus S| = |ao|$ and $|oa \setminus S| = |oa| - |S|$. Furthermore, $|oab| = |boa \cap S|$. Since $S \subseteq oa$ and $|oa| = |oab| + |oba| + |boa|$, we have

$$\begin{aligned} |oa \setminus S| &= |oba| + |boa \setminus S| \\ &= |oba| + |boa| - |boa \cap S| \\ &\leq |ba| - |boa \cap S| \\ &\leq |ab| - |oab| \leq |ao| = |ao \setminus S| \end{aligned}$$

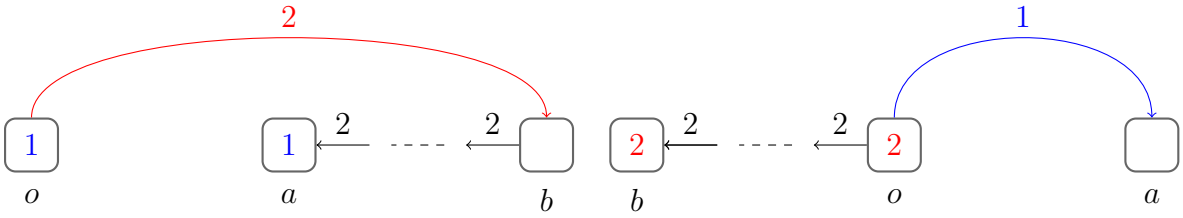
where the last inequality is because $ab \setminus oab = aob \cup abo \subseteq ao$.

The argument for the other case, where $|boa| < |oab|$, is similar. Define S to be a set containing all agents in boa and $|boa|$ agents from oab. Once again, $|ao \setminus S| = |ao|$ and $|oa \setminus S| = |oa| - |S|$. Furthermore, $|boa| = |oab \cap S|$. Since $S \subseteq oa$ and $|oa| = |oab| + |oba| + |boa|$, we have

$$\begin{aligned} |oa \setminus S| &= |oba| + |oab \setminus S| \\ &= |oba| + |oab| - |oab \cap S| \\ &\leq |ob| - |oab \cap S| \\ &\leq |bo| - |boa| \leq |ao| = |ao \setminus S| \end{aligned}$$

where the last inequality is because $bo \setminus boa = abo \cup bao \subseteq ao$. □

Let $S \subseteq \mathcal{C}$ be a set satisfying the conditions of Claim 3.1.3. Given such a set S, we can define a dual solution with $\gamma = 5$. We partition S into pairs $\{i_\ell, j_\ell\}$, where $i_\ell \in oab \cap S$ and $j_\ell \in boa \cap S$. Note that this is well-defined as $S \subseteq oab \dot\cup boa$ and $|S \cap oab| = |S \cap boa|$. The flow and charges in $\mathcal{N}_{i_\ell}$ and $\mathcal{N}_{j_\ell}$ are depicted in Figure 3.4.



(a) $\mathcal{N}_{i_\ell}$ (b) $\mathcal{N}_{j_\ell}$

Figure 3.4: Dual solution for $j \in S$

As shown in the above figure, the 2 units of flow sent along the $\alpha_{1,(i_\ell,o,b)}$-arc in $\mathcal{N}_{i_\ell}$ are charged to j_ℓ (i.e., $\alpha_{2,(j_\ell,b,o)} = 2$). Similarly, the 1 unit of flow sent along the $\alpha_{1,(j_\ell,o,a)}$-arc in $\mathcal{N}_{j_\ell}$ is charged to i_ℓ (i.e., $\alpha_{2,(i_\ell,o,a)} = 1$). Since $|ao \setminus S| \geq |oa \setminus S|$, we can use the construction given in Figure 3.3 for $j \notin S$. Note that the sum of the total charge and amount of flow departing from the source node is at most 5 in $\mathcal{N}_j$ for all $j \in S$, and is at most 3 for all $j \notin S$; hence, $\gamma = 5$, so $\mathrm{OPT}(D^\sigma_{ao}) \leq 5$.

□

3.1.2 Matching uncovered set

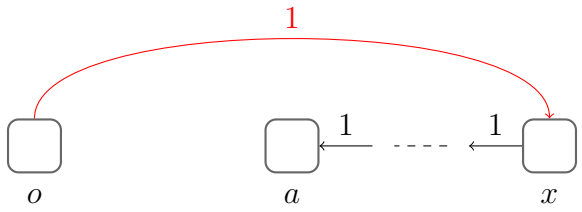
One important construction in the distortion literature is the *matching uncovered set* [37, 24, 28]. Munagala and Wang [37] proved that any candidate in the *matching uncovered set* is a 3-distortion candidate, and recently, Gkatzelis et. al proved that the matching uncovered set is not empty (thus resolving the deterministic optimal metric distortion conjecture). Construct a bipartite graph H_{ao} on a node set consisting of two disjoint copies of $\mathcal{C}$, which form the bipartition. There is an edge (i, j) in H_{ao} if and only if there exists $x \in A$ such that $a \succeq_i x$ and $x \succeq_j o$. Then, the matching uncovered set is precisely the set of all $a \in A$ such that there exists a perfect matching in H_{ab} for all $b \in A$.

Since our distortion upper bounds are constructed using a relaxation of (Q^{σ}_{ao}), it is pertinent to ask whether a candidate a in the matching-uncovered set also satisfies $\text{OPT}((P^{\sigma}_{ao})) \leq 3$ for all $o \in A$. We give a simple proof that this is indeed the case.

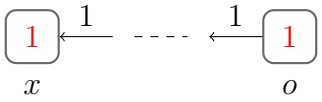
Theorem 3.1.4. *Let a be a candidate in the matching uncovered set. Then, $OPT(P^{\sigma}_{ao}) \leq 3$ for all $o \in A$.*

Proof. Since a is a candidate in the matching uncovered set, for any $o \in A$, H_{ao} has a perfect matching. Fix any $o \in A$ and let M be a perfect matching in H_{ao}. We will construct a dual solution with $\gamma = 3$. Notice that for every agent i, there is a copy of i on the "left" as well as on the "right". We show a separate flow for each copy of i on the left and on the right, and the final flow in $\mathcal{N}_i$ will be the sum of these flows. Furthermore, the flow for the left-copy of i, may use an $\alpha_{1,\cdot}$ arc, and charge this usage to a right-copy of a different agent. The flow for the right-copy of i may be used to satisfy an $\alpha_{2,\cdot}$ charge on i.

This is illustrated by the following concrete example. Let M be a perfect matching of H_{ao} and let $(i, j) \in M$, where i is a node on the left of H_{ao} and j is a node on the right. There exists some $x \in A$ such that $a \succeq_i x$ and $x \succeq_j o$. Then, the flow for the left-copy of i is given in Figure 3.5a and the flow for the right-copy of j is given in Figure 3.5b.



(a) Flow for left-copy of i in $\mathcal{N}_i$



(b) Flow for right-copy of j in $\mathcal{N}_j$

Figure 3.5: Dual solution for Theorem 3.1.4

Applying this construction to all $(i, j) \in M$, and taking the sum of the left and right flows in $\mathcal{N}_i$ for all $i \in \mathcal{C}$ yields a dual solution with $\gamma = 3$, so $\text{OPT}(D^{\sigma}_{ao}) \leq 3$. So, if H_{ao} has a perfect matching, $\text{OPT}(P^{\sigma}_{ao}) \leq 3$. Since a is in the matching uncovered set, H_{ao} has a perfect matching for all $o \in A$, so $\text{OPT}(P^{\sigma}_{ao}) \leq 3$ for all $o \in A$. □

As a corollary, if a is a candidate that could be chosen by Gkatzelis et. al's social choice function (PluralityMatching) [24], $\text{OPT}(P^{\sigma}_{ao}) \leq 3$ for all $o \in A$. However, the converse is not true; as shown by the following example, it is possible for a to have $\text{OPT}(P^{\sigma}_{ao}) \leq 3$ for all $o \in A$, but not be in the matching-uncovered set.

Example 3.2. Consider the following instance with four agents $\{1, 2, 3, 4\}$ and three candidates $\{a, b, o\}$, where the preference profile for agent 1 is $o \succeq a \succeq b$, for agents 2 and 3 is $b \succeq o \succeq a$, and for agent 4 is $a \succeq b \succeq o$. Figure 3.6 gives a feasible solution to (D^{σ}_{ao}), so $\text{OPT}(P^{\sigma}_{ao}) \leq 3$. Since $|ab| = |ba|$, by Fact 3.1.1, $\text{OPT}(P^{\sigma}_{ab}) \leq 3$, so a is a 3-distortion candidate. However, H_{ao} does not have a perfect matching (as $\{2, 3\}$ is a deficient set) so a is not in the matching uncovered set.

3.1.3 A sufficient condition for $\mathbf{OPT}(P^{\sigma}_{ao}) \leq 3$

As noted above, there can be a candidate a outside the matching-uncovered set for which we still have $\max_{o \in A} \mathrm{OPT}(P^{\sigma}_{ao}) \leq 3$. So it is useful to try to understand when such a bound applies. In this section, we use our analysis framework to derive a general sufficient condition for this. Later, we show that when $|A| \leq 3$, this condition is also necessary. (Thus, when $|A| \leq 3$, we obtain a tight characterization of alternatives a for which $\max_{o \in A} \mathrm{OPT}(P^{\sigma}_{ao}) \leq 3$).

Condition $(*)$: There exists $b \in A \setminus \{a, o\}$, $|ao| \geq \max\{|boa| - |oab|, |oab|\} + |oba|$.

Lemma 3.1.5. *If Condition* $(*)$ *holds, then* $OPT(P^{\sigma}_{ao}) \leq 3$.

Proof. Suppose Condition $(*)$ holds. Then, there exists $b \in A \setminus \{a, o\}$ such that $|ao| \geq 2\max\left\{\frac{|boa|}{2} - |oab|, 0\right\} + |oab| + |oba|$. We will prove that $\mathrm{OPT}(P^{\sigma}_{ao}) \leq 3$ by constructing a feasible solution to (D^{σ}_{ao}) of value at most 3.

Case 1: $|oab| \geq \frac{|boa|}{2}$. Since $|oab|$ is an integer, $|oab| \geq \left\lceil \frac{|boa|}{2} \right\rceil$. Define S to be a set containing boa, $\left\lceil \frac{boa}{2} \right\rceil$ agents from oab, and $\left\lceil \frac{boa}{2} \right\rceil$ agents from ao (note that this is well-defined as $\left\lceil \frac{boa}{2} \right\rceil \leq |oab| \leq |ao|$). For now, assume $|boa|$ is even. We can partition S into disjoint groups of 4 agents $\{i, j, k_1, k_2\}$ where $i \in oab \cap S$, $j \in ao \cap S$, $k_1, k_2 \in boa \cap S$.

In Figure 3.6, the colours indicate which nodes the $\alpha_{1,T}$-arc usage has been charged to. For instance, the cost of sending 1 unit of flow along the $\alpha_{1,(k_1,o,a)}$ arc is charged to $\mathcal{N}_i$, as $\alpha_{2,(i,o,a)} = 1$ (Figures 3.6a and 3.6b). Similarly, the cost of routing 2 units along the $\alpha_{1,(i,o,b)}$-arc is charged to the b and o nodes for agents k_1 and k_2. If $|boa|$ is odd, there will be one group of 3 agents, $\{i, j, k\}$ with $i \in oab \cap S$, $j \in ao \cap S$, $k \in boa \cap S$. In this case, the flow for this last group will be as depicted in Figure 3.7.

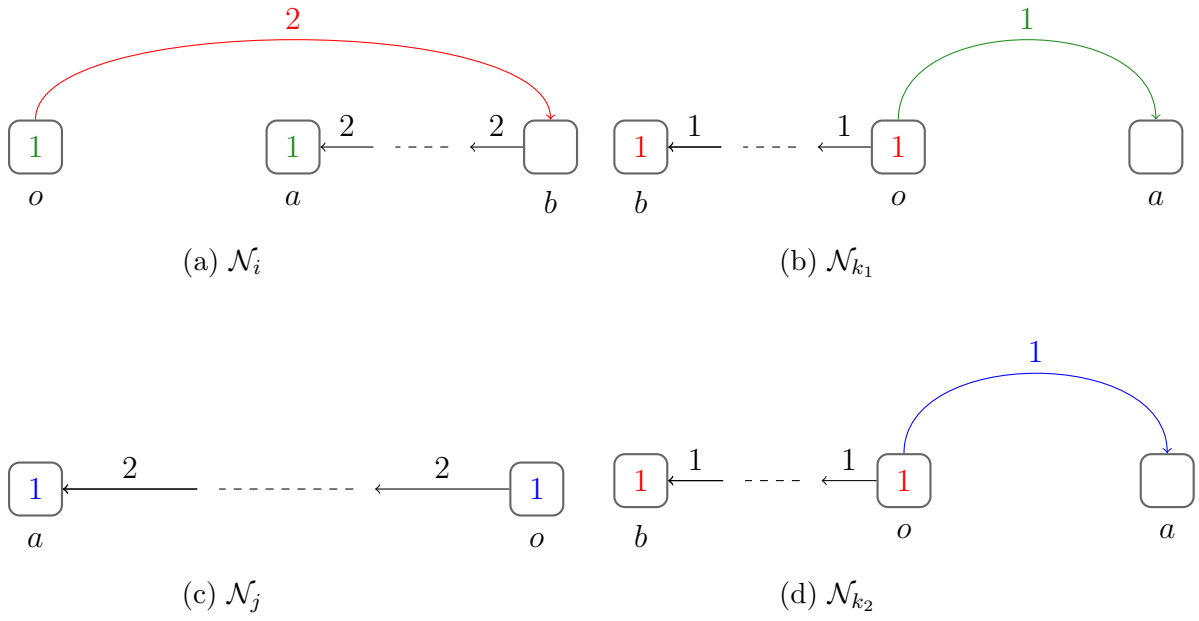


Figure 3.6: Partial dual solution for $\{i, j, k_1, k_2\}$

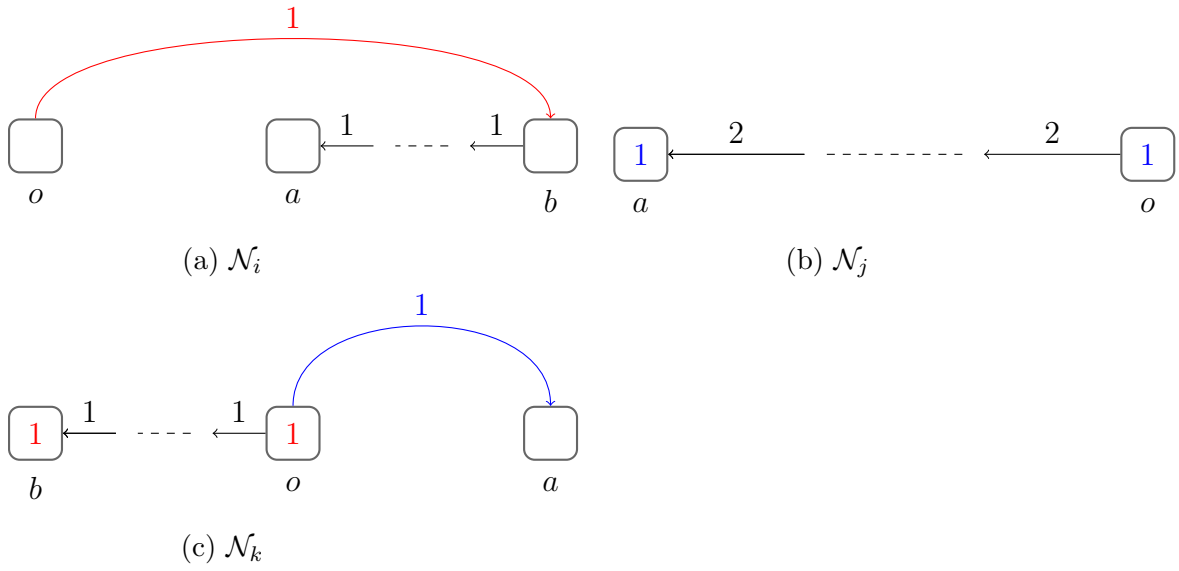


Figure 3.7: Partial dual solution for $\{i, j, k\}$ (if $|boa|$ is odd)

Notice that the total cost incurred by o in $\mathcal{N}_\ell$ for any agent $\ell \in S$ is at most 3, so we can construct a dual solution with $\gamma = 3$ when restricted to S. Moreover, as we show below, we have $|oa \setminus S| \leq |ao \setminus S|$, so we can use the construction given in Fact 3.1.1 for $\mathcal{C} \setminus S$. We have

$$\begin{aligned}
|oa \setminus S| &= |oba| + |oab \setminus S| + |boa \setminus S| \\
&= |oba| + |oab| - |oab \cap S| && \text{as } boa \subseteq S \\
&\leq |ao| - |oab \cap S| && \text{as } |ao| \geq |oab| + |oba| \\
&= |ao| - |ao \cap S| = |ao \setminus S|
\end{aligned}$$

Case 2: $|oab| < \frac{|boa|}{2}$. Then, $|ao| \geq |boa| + |oba| - |oab|$ Define S to be a set containing oab, $2|oab|$ agents from boa, and $|oab|$ agents from ao (note that this is well defined as $2|oab| < |boa|$ and $|oab| \leq |boa| - |oab| \leq |ao|$). As before, we can partition S into disjoint groups of 4 agents $\{i, j, k_1, k_2\}$ where $i \in oab \cap S$, $j \in ao \cap S$, $k_1, k_2 \in boa \cap S$, and use the same strategy as depicted in Figure 3.6. Thus, we can construct a dual solution with $\gamma = 3$ when restricted to S. Furthermore, as before we show that $|oa \setminus S| \leq |ao \setminus S|$, and hence we can use the construction given in Fact 3.1.1 for $\mathcal{C} \setminus S$. We have

$$\begin{aligned}
|oa \setminus S| &= |oba| + |boa \setminus S| + |oab \setminus S| \\
&= |oba| + |boa| - |boa \cap S| && \text{note that } oab \subseteq S \\
&\leq |ao| + |oab| - |boa \cap S| && \text{as } |ao| \geq |boa| + |oba| - |oab| \\
&= |ao| - |oab| \\
&= |ao| - |ao \cap S| = |ao \setminus S|
\end{aligned}$$

□

Surprisingly, Condition $(*)$ is in fact necessary when $m = 3$. We prove this by constructing primal solutions to (P^{σ}_{ao}). While Condition $(*)$ per se does not generalize beyond $m = 3$, it is plausible that this style of argument can be useful in other situations. In the sequel, we assume the set of candidates is $A = \{a, o, b\}$.

Lemma 3.1.6. *Suppose $m = 3$, with $A = \{a, o, b\}$, and $OPT(P^{\sigma}_{ao}) \leq 3$. Then, Condition $(*)$ holds, i.e., we have $|ao| \geq \max\{|boa| - |oab|, |oab|\} + |oba|$*

Proof. Case 1: $\frac{|boa|}{2} - |oab| < 0$. In this case, we wish to show that $|ao| \geq \max\{|boa| - |oab|, |oab|\} + |oba| = |oab| + |oba|$. Consider the metric $\tilde{d}$ defined by following graph. The squares denote candidates and the dark circles indicate the location of all agents in the specified set. The distance between a pair of agents/candidates is the length of the shortest path in the graph.

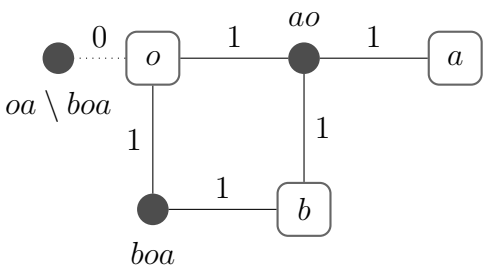


Figure 3.8: Metric $\tilde{d}$ when $2|oab| > |boa|$

Scaling the distances by $\frac{1}{|ao|+|boa|}$ yields a feasible solution to (P^{σ}_{ao}). The value of this solution is

$$\sum_{j \in \mathcal{C}} d_{aj} = |ao| \cdot \frac{1}{|ao| + |boa|} + |boa| \cdot \frac{3}{|ao| + |boa|} + |oa \setminus boa| \cdot \frac{2}{|ao| + |boa|} = 1 + 2 \cdot \frac{|oa|}{|ao| + |boa|}$$

If $\text{OPT}(P^{\sigma}_{ao}) \leq 3$ then the value of any primal solution must be at most 3, so we obtain the following necessary condition.

$$1 + \frac{2 \cdot |oa|}{|ao| + |boa|} \leq 3 \implies |oa| \leq |ao| + |boa|$$

Equivalently, $|ao| \geq |oa| - |boa| = |oab| + |oba|$.

Case 2: $\frac{|boa|}{2} - |oab| \geq 0$. In this case, we will show that $|ao| \geq 2\max\left\{\frac{|boa|}{2} - |oab|, 0\right\} + |oab| + |oba| = |boa| - |oab| + |oba|$. Consider the metric $\tilde{d}$ defined by following graph. As before, the squares denote candidates and the dark circles indicate the location of all agents

in the specified set. The distance between a pair of agents/candidates is the length of the shortest path in the graph.

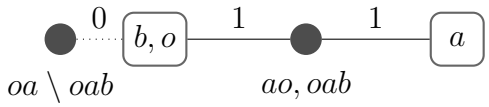


Figure 3.9: Metric $\tilde{d}$ when $2|oab| \leq |boa|$

Scaling the distances by $\frac{1}{|ao|+|oab|}$ yields a feasible solution to (P^{σ}_{ao}). The value of this solution is

$$\sum_{j \in \mathcal{C}} d_{aj} = (|ao| + |oab|) \cdot \frac{1}{|ao| + |oab|} + |oa \setminus oab| \cdot \frac{2}{|ao| + |oab|} = 1 + 2 \cdot \frac{|oa \setminus oab|}{|ao| + |oab|}$$

If $\text{OPT}(P^{\sigma}_{ao}) \leq 3$ then the value of any solution to (P^{σ}_{ao}) must be at most 3 so the following is a necessary condition:

$$1 + 2 \cdot \frac{|oa \setminus oab|}{|ao| + |oab|} \leq 3 \implies |ao| + |oab| \geq |oa \setminus oab|$$

Equivalently, $|ao| \geq |boa| + |oba| - |oab|$. □

3.2 Randomized social choice functions

So far, we have only considered deterministic social choice functions. However, it has been shown that, for the metric single winner determination problem, randomized social choice functions (SCFs) can be strictly more powerful than deterministic mechanisms [8, 24, 29]. In particular, Gkatzelis et. al [24] give a randomized SCF whose distortion is $3 - \frac{2}{m}$. A long-standing conjecture is that there exists a randomized social choice function that has a worst-case distortion of at most 2.

We develop a linear program that computes an instance-optimal randomized social choice function, i.e., a distribution that achieves minimum distortion for a given instance.

Using this, we present a simple instance, where the optimal distortion achievable by randomized SCFs is strictly larger than 2 (roughly 2.063164) thereby disproving the above conjecture. This result appears as the ArXiv paper [40]. This conjecture has also been refuted independently and concurrently by Charikar and Ramakrishnan [22], who also give a slightly tighter lower bound. However, our techniques differ significantly from theirs and, in particular, our LP for computing an instance-optimal randomized social choice function may be of independent interest.

A randomized social choice function f maps a preference profile σ to a distribution over the candidates A. For a randomized social choice function f, we define

$$\text{distortion}(f) = \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\mathbb{E}\left[\sum_{j \in \mathcal{C}} d(f(\sigma), j)\right]}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(o, j)}$$

3.2.1 Computing an instance-optimal randomized SCF

In this section we give a linear program which, given a preference profile σ, computes an instance-optimal randomized social choice function. In order to do so, we first consider the adversary's problem: Given a preference profile σ, randomized SCF f, and optimal candidate o, the adversary wishes to compute a metric d that is consistent with σ and maximizes $\frac{\mathbb{E}[\sum_{j \in \mathcal{C}} d(f(\sigma), j)]}{\sum_{j \in \mathcal{C}} d(o, j)}$. This can be done by solving the linear program (R^{σ}_{qo}), where q is the distribution over the candidates specified by $f(\sigma)$. As before, $m = |A|$, $n = |\mathcal{C}|$, and we use $\text{alt}_{\sigma}(k, r)$ to denote the rth ranked outcome in $\succeq_k \in \sigma$.

$$\max \quad \sum_{i \in A} \sum_{j \in \mathcal{C}} q_i d_{ij} \tag{R^{σ}_{qo}}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{C}} d_{oj} \leq 1 \tag{3.18}$$

$$d_{\text{alt}_{\sigma}(j,r),j} \leq d_{\text{alt}_{\sigma}(j,r+1),j} \qquad \forall j \in \mathcal{C},\ \forall r \in [m-1] \tag{3.19}$$

$$d_{ij} \leq d_{ik} + d_{jk} \qquad \forall i, j, k \in \mathcal{C} \cup A \tag{3.20}$$

$$d \geq 0 \tag{3.21}$$

Constraint (3.18) normalizes $\sum_{j\in\mathcal{C}} d_{oj}$ (allowing us to avoid having a ratio in the objective). The remaining constraints are precisely constraints (3.2)-(3.9) in (Q^{σ}_{ao}); constraint (3.19) ensures that the metric d is consistent with the preference profile σ, and constraints (3.20)-(3.21) enforce that d is a metric. The optimal solution d is a metric that maximizes $\frac{\mathbb{E}[\sum_{j\in\mathcal{C}} d_{f(\sigma)j}]}{\sum_{j\in\mathcal{C}} d_{oj}} = \frac{\sum_{i\in\mathcal{F}}\sum_{j\in\mathcal{C}} q_i d_{ij}}{\sum_{j\in\mathcal{C}} d_{oj}}$.

We now return to our original task, which is to compute an instance-optimal randomized SCF. Equivalently, we wish to compute an optimal distribution $q \in \Delta_A$ that minimizes $\max_{o\in A} \mathrm{OPT}(R^{\sigma}_{qo})$. Notice that (R^{σ}_{qo}) is of the form $\min_d\{c^T d : A^o d \leq b^o, d \geq 0\}$, where c depends linearly on q and only A^o and b^o depend on the choice of o. The dual of (R^{σ}_{qo}) is $(\tilde{D}^{\sigma}_{qo})$: $\min_y\{y^T b^o : y^T A^o \geq c, y \geq 0\}$. So, the problem of computing an instance-optimal randomized SCF is equivalent to

$$\begin{aligned}\min_{q\in\Delta_A}\max_{o\in A}\mathrm{OPT}(R^{\sigma}_{qo}) &= \min_{q\in\Delta_A}\min\{\gamma : \mathrm{OPT}(R^{\sigma}_{qo}) \leq \gamma\ \forall o\in A\}\\ &= \min_{q\in\Delta_A}\min\{\gamma : \forall o\in A\ \exists y^o \geq 0 \text{ s.t. } (y^o)^T A^o \geq c^T, (y^o)^T b^o \leq \gamma\}\end{aligned}$$

where the last equality follows due to LP-duality. Note that $c = Hq$ above, where H is a matrix whose rows are indexed by $A\times\mathcal{C}$ and whose columns are indexed by A: we have $H_{ij,i} = q_i$ for all $i \in A, j \in \mathcal{C}$. Thus, we obtain the following LP for finding an instance-optimal randomized SCF.

$$\begin{aligned}
\min\ & \gamma && && \text{(Best-Dist)}\\
\text{s.t. }& (y^o)^T A^o - q^T H^T \geq 0 && \forall o\in A && (3.22)\\
& \gamma - (y^o)^T b^o \geq 0 && \forall o\in A && (3.23)\\
& \sum_{i\in A} q_i \geq 1 && && (3.24)\\
& y^o, q \geq 0 && \forall o\in A && (3.25)
\end{aligned}$$

(Note that in (Best-Dist), we have replaced $\sum_i q_i = 1$ with an inequality. This is inconsequential: it is easy to see that one may assume that an optimal solution satisfies the inequality tightly).

The interested reader can find the explicit expansion of (Best-Dist) in Appendix A. For the purposes of taking the dual of (Best-Dist), it will however be much easier to retain the above form. Let $d^o \in \mathbb{R}_+^{A \times \mathcal{C}}$ denote the dual variables corresponding to (3.22), $w^o \in \mathbb{R}^+$ be the dual variables corresponding to (3.23), and φ denotes the dual variable corresponding to (3.24). The dual of (Best-Dist) is then

$$\begin{aligned}
\max \quad & \varphi \\
\text{s.t.} \quad & A^o d^o - b^o w^o \leq 0 \qquad \forall o \in A && (3.26)\\
& \sum_o w^o \leq 1 && (3.27)\\
& \varphi e - H^T(\sum_{o \in A} d^o) \leq 0 && (3.28)\\
& \varphi, d^o, w^o \geq 0 \qquad \forall o \in A && (3.29)
\end{aligned}$$

This LP, and hence the dual of (Best-Dist), is equivalent to

$$\begin{aligned}
\max \quad & \varphi && && \text{(Best-Dist-Dual)}\\
\text{s.t.} \quad & \sum_{o \in A} \sum_{j \in \mathcal{C}} d^o_{oj} \leq 1 && && (3.30)\\
& \varphi - \sum_{o \in A} \sum_{j \in \mathcal{C}} d^o_{ij} \leq 0 && \forall i \in A && (3.31)\\
& d^o_{\mathrm{alt}_\sigma(j,r),j} \leq d^o_{\mathrm{alt}_\sigma(j,r+1),j} && \forall j \in \mathcal{C}\, \forall r \in [m-1] \forall o \in A && (3.32)\\
& d^o_{ij} \leq d^o_{ik} + d^o_{jk} && \forall i,j,k \in \mathcal{C} \cup A\, \forall o \in A && (3.33)\\
& \varphi, d^o \geq 0 && \forall o \in A && (3.34)
\end{aligned}$$

3.2.2 A lower bound for the distortion of randomized SCFs

We show that, for any randomized social choice function f, $\mathrm{distortion}(f) \geq 2.063164$.

Theorem 3.2.1. *There exists an instance $(\mathcal{C}, A, \sigma)$ such that, for any randomized social choice function f, $\sup_{d \lhd \sigma} \frac{\mathbb{E}[\sum_{j \in \mathcal{C}} d(f(\sigma), j)]}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(o,j)} \geq 2.063164$.*

Proof. We show this by designing an instance and a feasible solution to (Best-Dist-Dual) of objective value at least 2.063164.

Consider the following instance with $\mathcal{C} = \{1, 2, 3, 4, 5, 6, 7\}$ and $A = \{a, b, c, d, e, f, g\}$. The preference profile σ is given by

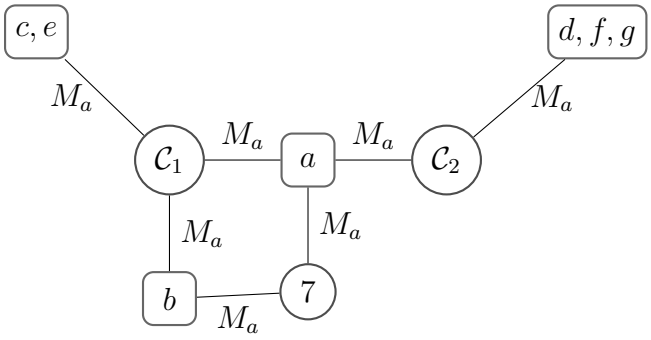
$$1, 2, 3 : c \succeq e \succeq b \succeq a \succeq f \succeq g \succeq d, \qquad 4, 5, 6 : d \succeq g \succeq f \succeq a \succeq e \succeq b \succeq c,$$
$$7 : b \succeq a \succeq f \succeq g \succeq e \succeq c \succeq d$$

For this instance, OPT(Best-Dist) = 2.063164 – that is, for any randomized social choice function f, $\sup_{d \triangleleft \sigma} \frac{\mathbb{E}[\sum_{j \in \mathcal{C}} d_{f(\sigma)j}]}{\min_{o \in \mathcal{F}} \sum_{j \in \mathcal{C}} d_{oj}} \geq 2.063164$. The instance-optimal distribution q is

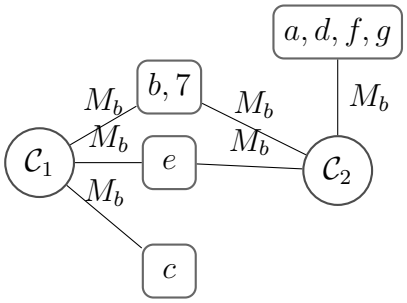
$$q_a = 0.039301 \qquad q_b = 0.121723 \qquad q_c = 0.388299 \qquad q_d = 0.291224$$
$$q_e = 0.107872 \qquad q_f = 0.029475 \qquad q_g = 0.022107$$

We can verify that OPT(Best-Dist) ≤ 2.063164 via the following solution to the dual (Best-Dist-Dual), which has value 2.063164. The dual variables $d^a, d^b, \ldots, d^g$ can be interpreted as metrics, which are represented by the graphs given below. In this solution, agents with the same preference rankings are colocated – namely, agents in $\mathcal{C}_1 = \{1, 2, 3\}$ are colocated and agents in $\mathcal{C}_2 = \{4, 5, 6\}$ are colocated. □

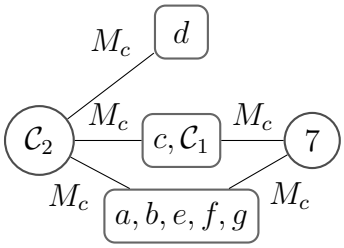
Figure 3.10: An optimal solution to the dual of (Best-Dist)



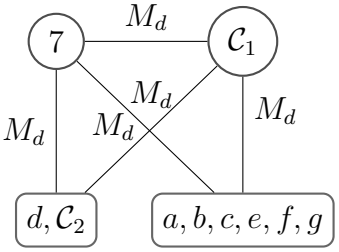
(a) For any $i, j \in \mathcal{C} \cup A$, d^a_{ij} is the shortest-path distance in the above graph, where $M_a = 0.014507$



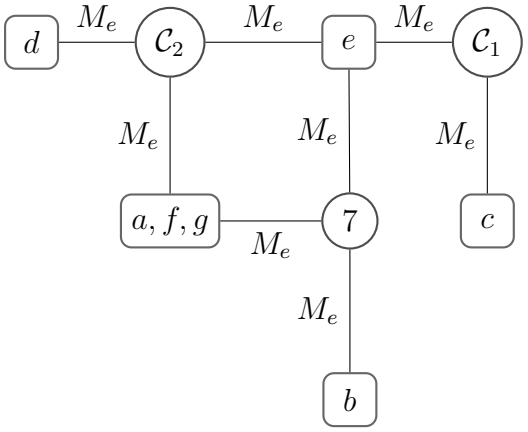
(b) For any $i, j \in \mathcal{C} \cup A$, d^b_{ij} is the shortest-path distance in the above graph, where $M_b = 0.020955$



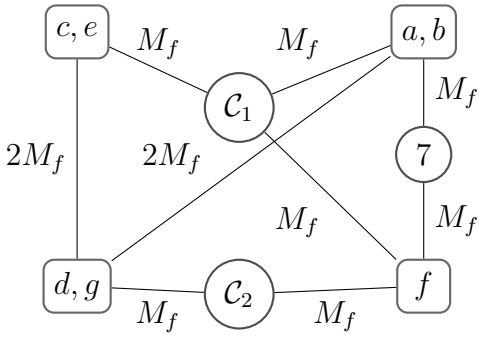
(c) For any $i, j \in \mathcal{C} \cup A$, d^c_{ij} is the shortest-path distance in the above graph, where $M_c = 0.038866$



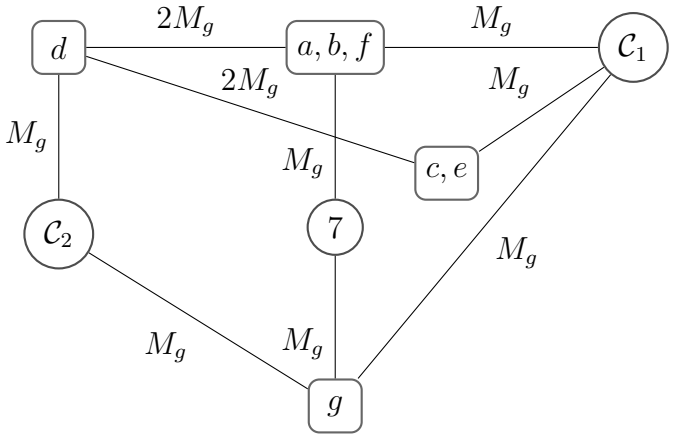
(d) For any $i, j \in \mathcal{C} \cup A$, d^d_{ij} is the shortest-path distance in the above graph, where $M_d = 0.051820$



(e) For any $i, j \in \mathcal{C} \cup A$, d^e_{ij} is the shortest-path distance in the above graph, where $M_e = 0.013433$



(f) For any $i, j \in \mathcal{C} \cup A$, d^f_{ij} is the shortest-path distance in the above graph, where $M_f = 0.019343$



(g) For any $i, j \in \mathcal{C} \cup A$, d^g_{ij} is the shortest-path distance in the above graph, where $M_g = 0.025791$

3.2.3 Extending the analysis framework to randomized SCFs

By the same argument given in section 3.1, in order to prove that distortion$(f) \leq \rho$ for a randomized social choice function f, it would suffice to show that $(R^{\sigma}_{f(\sigma),o})$ has an optimal value of at most ρ for all $o \in A$, as this would imply that $\sup_{d \triangleleft \sigma} \frac{\mathbb{E}[\sum_{j \in \mathcal{C}} d(f(\sigma),j)]}{\min_{o \in A} \sum_{j \in \mathcal{C}} d(o,j)} \leq \rho$. We could show this by demonstrating a dual solution of value at most ρ, however, (Best-Dist) is difficult to interpret. Instead, as before, we will consider the relaxation of (R^{σ}_{qo}) obtained by dropping all triangle-inequality constraints (3.20) that do not correspond to triangles in $\mathcal{T}(\sigma, o) := \{(j, i_1, i_2) : i_1, i_2 \in A : i_1 \neq i_2, i_1 \succeq_j i_2, o \in \{i_1, i_2\}, j \in \mathcal{C}\}$. This yields the following LP.

$$\max \quad \sum_{i \in A} \sum_{j \in \mathcal{C}} q_i d_{ij} \tag{P^{σ}_{qo}}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{C}} d_{oj} \leq 1 \tag{3.35}$$

$$d_{\text{alt}_\sigma(j,r),j} \leq d_{\text{alt}_\sigma(j,r+1),j} \qquad \forall j \in \mathcal{C}\, \forall r \in [m-1] \tag{3.36}$$

$$d_{i_2 j} \leq d_{i_1 j} + d_{i_1 i_2} \qquad \forall (j, i_1, i_2) \in \mathcal{T}(\sigma, o) \tag{3.37}$$

$$d_{i_1 i_2} \leq d_{i_1 j} + d_{i_2 j} \qquad \forall (j, i_1, i_2) \in \mathcal{T}(\sigma, o) \tag{3.38}$$

$$d \geq 0 \tag{3.39}$$

Note that the constraints of (P^{σ}_{qo}) are precisely the constraints of (P^{σ}_{ao}); it is only the objective function that is different. Consequently, the dual of (P^{σ}_{qo}), (D^{σ}_{qo}), is very similar to (D^{σ}_{ao}), and the only change is that the RHS of (3.40) is now q_i instead of 1.

$$\min \quad \gamma \tag{D^{σ}_{qo}}$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{T=(k,\cdot,i)\in\mathcal{T}} (\alpha_{1,T} - \alpha_{2,T}) - \sum_{T=(k,i,\cdot)\in\mathcal{T}} (\alpha_{1,T} + \alpha_{2,T}) + \beta_{1k}\mathbb{I}_{[i=\text{alt}_\sigma(k,1)]} \\
& \quad + \sum_{r=1}^{n-2} (\beta_{r+1,k} - \beta_{r,k})\mathbb{I}_{[i=\text{alt}_\sigma(k,r+1)]} - \beta_{n-1,k}\mathbb{I}_{[i=\text{alt}_\sigma(k,n)]} \\
& \quad + \gamma\mathbb{I}_{[i=o]} \geq q_i \qquad \forall k \in \mathcal{C}, \forall i \in A \qquad (3.40) \\
& \sum_{T\in\mathcal{T}:T=(\cdot,i_1,i_2) \text{ or } (\cdot,i_2,i_1)} (\alpha_{2,T} - \alpha_{1,T}) \geq 0 \qquad \forall i_1, i_2 \in A \qquad (3.41) \\
& \alpha, \beta, \gamma \geq 0 \qquad (3.42)
\end{aligned}$$

As before, a feasible solution to (D^{σ}_{qo}) can be interpreted as a flow in $\mathcal{N}_j$ (Figure 3.2). The only difference between a feasible solution to (D^{σ}_{qo}) and a feasible solution to (D^{σ}_{ao}) is that every node i now has a demand of q_i (whereas in the deterministic setting, the only sink was the node corresponding to a, which had a demand of 1). Thus, the analysis framework presented in Section 3.1 extends readily to randomized social choice functions.

Chapter 4

Multiwinner selection and the k-median problem

Given the existence of a 3-distortion social choice function when electing a single winner, a natural question to ask is whether similar results can be obtained if one wishes to elect a committee of k candidates, for any $k \geq 2$. More formally, in the k-winner selection problem, we are given a set of agents $\mathcal{C}$ and candidates (or alternatives) A. We assume that the agents and candidates lie in a metric space $(\mathcal{C} \cup A, d)$, where $d : (\mathcal{C} \cup A)^2 \to \mathbb{R}_{\geq 0}$ is a distance function satisfying the triangle inequality. The cost incurred by an agent $j \in \mathcal{C}$ when a committee $S \subseteq A$ is elected is $d(j, S) = \min_{s \in S} d(j, s)$. Our objective is to choose a committee S consisting of k winners from A so as to minimize $\sum_{j \in \mathcal{C}} d(j, S)$. As discussed in Chapter 2, we can extend the definition of distortion to this setting; given a social choice k-correspondence $f : L \to [A]^k$ (where L is the set of total orders on A and $[A]^k$ denotes the k-subsets of A), the distortion of f is

$$\text{distortion}(f) = \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} d(j, f(\sigma))}{\min_{S \subseteq [A]^k} \sum_{j \in \mathcal{C}} d(j, S)}$$

For the rest of this thesis, we will restrict our attention to peer-selection, wherein $A = \mathcal{C}$. Anshelevich and Zhu showed that, when $k = 2$, any social choice k-correspondence has a distortion of at least $\Omega(n)$ [10]. As proved by the following theorem, even in this restricted

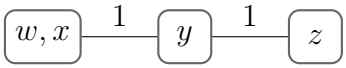
setting there does not exist a social choice k-correspondence with bounded distortion when $k \geq 3$. We note that this result has also been proved independently by Caragiannis, Shah, and Voudouris [18].

Theorem 4.0.1. *For $k \geq 3$, there exists an instance $(\mathcal{C}, A, \sigma)$ with $A = \mathcal{C}$ such that any social choice k-correspondence for the k-winner selection problem has unbounded distortion.*

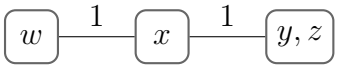
Proof. Consider the following instance with four clients where $A = \mathcal{C} = \{w, x, y, z\}$. The preference rankings are

$$w : x \succeq y \succeq z \quad x : w \succeq y \succeq z \quad y : z \succeq x \succeq w \quad z : y \succeq x \succeq w$$

The following metrics d_1 and d_2 are consistent with this preference ranking:



(a) For any $i, j \in \mathcal{C}$, $d_1(i, j)$ is the shortest path distance in the above graph.



(b) For any $i, j \in \mathcal{C}$, $d_2(i, j)$ is the shortest path distance in the above graph.

Figure 4.1: A k-winner selection instance with unbounded distortion

The optimal solution when considering d_1 is to choose $\{x, y, z\}$ as our committee – this solution incurs a social cost of 0. Moreover, any other committee incurs a social cost of at least 1. On the other hand, the optimal solution under d_2 is to choose $\{w, x, y\}$ as our committee. This solution incurs a social cost of 0, and any other solution incurs a social cost of at least 1 (with respect to d_2). Since the (ordinal) information provided to us is insufficient to differentiate between d_1 and d_2, the distortion of any social choice k-correspondence is unbounded on this instance. □

Remark 4.0.2. Consider the problem of computing a minimum cost k-Forest in the metric space $(\mathcal{C}, d)$. A purely-ordinal algorithm for this problem takes $(\mathcal{C}, \sigma)$ as input, and yields a forest consisting of k components. The counterexample given in the proof of Theorem 4.0.1 also illustrates that the distortion of any purely-ordinal algorithm for the k-Forest problem is also unbounded, for $k \geq 3$.

The upshot of the above discussion is that a good mechanism for the k-winner selection problem that has bounded distortion cannot rely only upon ordinal information, and must use some additional cardinal information.

4.1 Two k-median algorithms

As noted earlier, if we knew the true underlying metric, the k-winner selection problem described above is in fact equivalent to the *k-median* clustering problem, wherein one has a set of clients $\mathcal{C}$, facilities $\mathcal{F}$ (which corresponds to the set of alternatives, A), and wishes to open a set of k centers in $\mathcal{F}$, S, so as to minimize $\sum_{j \in \mathcal{C}} d(j, S)$. Due to this strong relationship between the two problems, we will refer to k-median and k-winner selection interchangeably throughout the rest of this thesis. The k-median problem is very well studied in the theoretical computer science literature (see, for instance, [14] and the references within). We will, however, restrict our attention to two algorithms that will be particularly important in the following chapters.

The first algorithm is a straightforward extension of Meyerson's online algorithm for facility location [35] (with uniform facility opening costs). In the facility location problem, there is no constraint on the number of facilities/centers that can be opened, but every facility has an opening cost f_i, which must be paid if the facility is selected.

Algorithm 1: Online facility location algorithm with uniform costs [35]

Data: Facility cost f, Sequence of clients $x_1, \ldots, x_n$

$S \leftarrow \{x_1\}$
for $i = 2, \ldots, n$ **do**
    $\delta_i = d(x_i, S)$
    Add x_i to S with probability $\min(1, \delta_i / f)$
end
return S

In the proof of Theorem 2.1, Meyerson [35] proves the following result.

Theorem 4.1.1 ([35]). *Let $x_1, \ldots, x_n$ be a sequence of clients in random order. Let $C_1^*, \ldots, C_k^*$ be the clusters induced by an optimal solution. Let S be the set of centers*

opened by Algorithm 1. Then, for every cluster C_i^*, $\mathbb{E}\left[|S \cap C_i^*| f + \sum_{j \in C_i^*} d(j,S)\right] \leq 5f + 8\sum_{j \in C_i^*} d(j, c_i^*)$.

We briefly give a sketch of Meyerson's analysis here, and defer the precise details of the proof to Chapter 6, where we modify Algorithm 1 to handle more general objectives. Let C^* be a cluster induced by the optimal solution. One can define the *core* of C^* to be the $|C^*|/2$ points closest to the cluster center. If, at any point, a center is opened within the core of C^*, the cost incurred by any other client in C^* under S is comparable to the cost it incurs under the optimal solution. Moreover, the expected cost incurred by the *core-clients* (i.e., clients in the core of C^*) before a center in the core of C^* is opened is at most f. Finally, $\mathbb{E}[\delta_b]$ for any client b that is not in the core of C^* can be bound in terms of $\mathbb{E}[\delta_g]$ (where g is the last core-client preceding b). Combining these claims yields the final bound on the expected cost of every cluster. An immediate corollary of this theorem is that the number of centers opened in each cluster is not too large either.

Corollary 4.1.2. *Let* $x_1, \ldots, x_n$ *be a sequence of clients in random order. Let* $C_1^*, \ldots, C_k^*$ *be the clusters induced by an optimal solution. Let* S *be the set of centers opened by Algorithm 1. Then, for every cluster* C_i^*, $\mathbb{E}\left[|S \cap C_i^*|\right] \leq 5 + 8 \cdot \frac{\sum_{j \in C_i^*} d(j, c_i^*)}{f}$.

An (α, β)-*bicriteria approximation algorithm* for the k-median problem yields a set of centers S such that $\sum_{j \in \mathcal{C}}(d(j,S)) \leq \alpha \cdot \mathrm{OPT}$, and $|S| \leq \beta k$. Algorithm 1 can be adapted to obtain a constant-factor bicriteria approximation algorithm for the k-median problem, by setting the facility opening cost to be $f = \frac{L}{k}$, where L is a $\Theta(1)$-estimate of the cost of an optimal k-median solution.

Corollary 4.1.3. *Let* $x_1, \ldots, x_n$ *be a sequence of clients in random order. Let* $C_1^*, \ldots, C_k^*$ *be the clusters induced by an optimal* k*-median solution. Let* S *be the set of centers opened by Algorithm 1 with* $f = \frac{L}{k}$. *Then,* $\mathbb{E}\left[\sum_{j \in \mathcal{C}} d(j,S)\right] \leq 5L + 8 \cdot OPT$, *and* $\mathbb{E}\left[|S|\right] \leq \left(5 + 8 \cdot \frac{OPT}{L}\right) k$.

Another elegant algorithm for k-means clustering and k-median clustering was given by Aggarwal, Deshpande, and Kannan [3]. Aggarwal et. al showed that a simple adaptive

sampling algorithm yields a $(O(1), O(1))$-bicriteria approximation for the k-means problem. As they observed in their paper, the algorithm extends readily to the k-median setting.

Algorithm 2: Adaptive sampling algorithm for k-median [3]

```
S_0 ← ∅
for i = 1, ..., τ(k + √k) do
    Sample s_i with probability proportional to d(s_i, S_{i-1})
    Update S_i ← S_{i-1} ∪ {s_i}.
end
Return S_{τ(k+√k)}
```

Theorem 4.1.4 (Aggarwal, Deshpande, and Kannan (2009))**.** *If $\tau = 16(k + \sqrt{k})$, Algorithm 2 yields a $\left(20, 16\left(1 + \frac{1}{\sqrt{k}}\right)\right)$-bicriteria approximation for the k-median problem, with constant probability.*

Once again, we will give a sketch of the proof here, and defer the precise argument to Chapters 6 and 7, where we extend the adaptive sampling algorithm to the ℓ-centrum and minimum-norm settings respectively. Let C^* be a cluster induced by the optimal solution. Once again, the *core* of C^* consists of the points that are close to the cluster center. We say that C^* is *good*, if the total cost incurred by the clients in C^* under the current solution is comparable to the cost they incur under the optimal solution (otherwise, we say that C^* is a *bad* cluster). As we have mentioned earlier, opening a center in the core of C^* causes the cost of all other clients in C^* to be small. That is, if C^* is currently a bad cluster, and we open a center in its core, then it will become a good cluster. The key insight of the analysis of Aggarwal et. al is that, as we are choosing the next center to open with probability proportional to the distance from the client to the currently open centers, either the cost of our current solution will be within some constant factor of OPT, or, with constant probability we will open a center in the core of some bad cluster. Then, by using concentration inequalities, one can argue that after $O(k)$ centers are opened, all clusters will be good with constant probability.

Chapter 5

k-median with limited value queries

In the previous chapter, we concluded our discussion of low-distortion algorithms for k-median by noting that any such algorithm must use some additional cardinal information (in addition to the preference profile σ). It will be helpful to recall the definition of a mechanism in this context.

Definition 2.1.3 (Mechanism [4])**.** A *mechanism* $\mathcal{M} = (\mathcal{Q}, f, k)$ with access to a query oracle takes as input a preference profile σ and returns a k-subset of A. It consists of an algorithm $\mathcal{Q}$ that, given an input preference profile σ, adaptively makes queries to the query oracle, and a function f that takes the input σ, the set of queries and their answers, and outputs a k-subset of A (i.e., a solution).

While different query models have enjoyed varying levels of success for some social-welfare maximization problems [4, 34], little is known for the social cost minimization setting. One of the most natural queries is to directly ask the agent for the exact distance between herself and given candidate. We refer to such queries as *value queries*. In this chapter, we give mechanisms that elicit a limited number of value queries from the agents (we assume that the preference profile σ is provided as input).

5.1 A blackbox reduction to the cardinal setting

If the true underlying metric is (approximately) known, this problem would be reduced to the cardinal k-median problem, which is a well-studied and well-understood problem. Hence, it would be ideal if we had techniques to leverage this understanding and export existing (cardinal) k-median algorithms to low-distortion mechanisms. As queries can be taxing on the agents, our reduction should require only a few queries per agent. Clearly, using $O(n)$ queries per agent, hence $O(n^2)$ queries in total, we can infer the underlying metric exactly, so our goal is to perform significantly fewer queries than this, namely $o(n)$ queries, or a query complexity that is independent of n. In this section, we give a blackbox reduction to the (cardinal) k-median problem using $polylog(n)$ queries per agent; later in this chapter, we will discuss how to improve its query complexity. We will consider a somewhat more general setting, where each agent comes with a positive integer weight, which denotes the number of co-located agents. Thus, the k-median cost of a set of centers S is $\sum_i w_i d(i, S)$, where $d(i, S)$ is the distance between i and the closest open center in S.

Let d^* denote the true underlying metric (which we do not know). Given an α-approximate estimate B of OPT, an upper bound on OPT(d^*), for any given point i, we consider all distances $d \geq \frac{B}{\alpha w_i n}$ and compute all points whose distance from i lies in $(d, (1+\varepsilon)]$. The idea is that we can approximate $d^*(i, j)$ by any distance in this interval, and this would only lose a $(1+\varepsilon)$-factor in the cost. For points with $d^*(i, j) \leq \frac{B}{\alpha w_i n}$, we can estimate their distance by any $d \leq \frac{B}{\alpha w_i n}$; this can incur an additive loss of at most $w_i \cdot \frac{B}{\alpha w_i n} = \frac{B}{\alpha n}$, when considering the total cost of the w_i points co-located with i. Thus, if we have any metric $\tilde{d}$ compatible with these estimates, then $\tilde{d}$ is "close enough" to d^* so that working with $\tilde{d}$ results in only a small loss, and so we can run our algorithm for cardinal k-median on $\tilde{d}$.

In the sequel, we will use OPT(d^*) to denote the value of an optimal solution with respect to d^*; when the metric is clear from the context, we will simply write OPT.

Algorithm 3: A blackbox reduction to k-median

Input: A set of n agents $\mathcal{C}$ with preference profile σ, and non-negative, integer weights $\{w_j\}_{j\in\mathcal{C}}$; d^* denotes the true underlying metric (which we do not know).

A constant B such that $B \in [\mathrm{OPT}(d^*), \alpha \cdot \mathrm{OPT}(d^*)]$

A ρ-approximation algorithm $\mathcal{A}$ for solving k-median on the weighted instance

for $i \in \mathcal{C}$ **do**
  Define $B_{i,0} = \rho(1+3\varepsilon) \cdot \frac{B}{w_i}$, $q_i := \left\lceil \log_{1+\varepsilon}(\frac{\alpha w_i B_{i,0} \cdot n}{\varepsilon B}) \right\rceil$
  Compute $S_{i,\ell} = \{j \in \mathcal{C} : d^*(i,j) \leq B_{i,0}(1+\varepsilon)^{-\ell}\}$ for $\ell = 0, \ldots, q_i$
end
Compute $\tilde{d}$ such that $\tilde{d}$ is a valid metric and also satisfies the following constraints:

(1) $\tilde{d}(i,j) \geq B_{i0}$ for all $j \notin S_{i,0}$.

(2) $(1+\varepsilon)^{-(\ell+1)} B_{i0} \leq \tilde{d}(i,j) \leq (1+\varepsilon)^{-\ell} B_{i0}$ for all $j \in S_{i,\ell} \setminus S_{i,\ell+1}$, $i \in \mathcal{C}, \ell \in \{0, \ldots, q_i - 1\}$

(3) $\tilde{d}(i,j) \leq \frac{\varepsilon B}{\alpha n \cdot w_i}$ for all $j \in S_{i,q_i}$

return $\mathcal{A}(\mathcal{C}, w, \tilde{d})$

Theorem 5.1.1. *Let d^* be the true underlying metric, and let F be the set of centers opened by Algorithm 3. Then,*

$$\sum_{j\in\mathcal{C}} w_j d^*(j,F) \leq (\rho(1+2\varepsilon)+\varepsilon) OPT(d^*)$$

Furthermore, Algorithm 3 can be implemented using $O(\log(n) \cdot \log(\alpha\rho \cdot n)/\varepsilon)$ value queries per agent.

Proof. Let $\mathrm{OPT}(\tilde{d})$ denote the value of an optimal k-median solution computed with respect to $\tilde{d}$. The following fact is immediate from the definition of $\tilde{d}$.

Fact 5.1.2. For any $i, j \in \mathcal{C}$, if $d^*(i,j) \le B_{i,0}$, then $d^*(i,j) - \kappa_i \le \tilde{d}(i,j) \le (1+\varepsilon)d^*(i,j) + \kappa_i$, where $\kappa_i = \frac{\varepsilon B}{\alpha w_i n}$.

Given this, we show that for a set of centers T, if $d^*(i,T) \le B_{i,0}$ for all $i \in \mathcal{C}$, the $\tilde{d}$-cost of T is a good approximation of the d^*-cost of T, and vice versa.

Claim 5.1.3. *Let $T \subseteq \mathcal{C}$ such that $d^*(i,T) \le B_{i,0}$ for all $i \in \mathcal{C}$. Then,*

(a) $\sum_{j\in\mathcal{C}} w_j \tilde{d}(j,T) \le (1+\varepsilon)\sum_{j\in\mathcal{C}} w_j d^*(j,T) + \varepsilon OPT(d^*)$

(b) $\sum_{j\in\mathcal{C}} w_j d^*(j,T) \le \sum_{j\in\mathcal{C}} w_j \tilde{d}(j,T) + \varepsilon OPT(d^*)$

Proof. Since $d^*(i,T) \le B_{i,0}$ for all $i \in \mathcal{C}$, by Fact 5.1.2,

$$\sum_{j\in\mathcal{C}} w_j d^*(j,T) - \sum_{j\in\mathcal{C}} w_j\kappa_j \le \sum_{j\in\mathcal{C}} w_j \tilde{d}(j,T) \le (1+\varepsilon)\sum_{j\in\mathcal{C}} w_j d^*(j,T) + \sum_{j\in\mathcal{C}} w_j\kappa_j$$

where $\kappa_i = \frac{\varepsilon B}{\alpha w_i n}$. Since $\frac{B}{\alpha} \le \mathrm{OPT}(d^*)$, $\sum_{j\in\mathcal{C}} w_j\kappa_j \le \varepsilon \mathrm{OPT}(d^*)$, so we obtain (a) and (b) as required. □

Let F^* be the set of centers opened by an optimal solution with respect to d^*. By Claim 5.1.3(a),

$$\mathrm{OPT}(\tilde{d}) \le \sum_{j\in\mathcal{C}} w_j \tilde{d}(j,F^*) \le (1+2\varepsilon)\cdot \mathrm{OPT}(d^*)$$

Let F be the set of centers opened by $\mathcal{A}(\mathcal{C}, w, \tilde{d})$. In order to use Claim 5.1.3, we must show that $d^*(j,F) \le B_{i,0}$ for all $j \in \mathcal{C}$. We prove that this property holds for any ρ-approximate solution (with respect to $\tilde{d}$).

Claim 5.1.4. *Let $T \subseteq \mathcal{C}$. If $\sum_{j\in\mathcal{C}} w_j \tilde{d}(j,T) \le \rho \cdot OPT(\tilde{d})$, $d^*(i,T) \le B_{i,0}$ for all $i \in \mathcal{C}$.*

Proof. Suppose, to arrive at a contradiction, that there exists $j \in \mathcal{C}$ such that $d^*(j,T) > B_{j,0}$. Then, $\tilde{d}(j,T) > B_{j,0}$ and hence

$$\sum_{i\in\mathcal{C}} w_i \tilde{d}(i,T) \ge w_j \cdot \tilde{d}(j,T) \ge w_j B_{j,0} > \rho(1+2\varepsilon)\mathrm{OPT}(d^*) \ge \rho\cdot \mathrm{OPT}(\tilde{d})$$

□

Recall that F is a ρ-approximate solution (with respect to $\tilde{d}$). As $\sum_{j\in\mathcal{C}} w_j \tilde{d}(j,F) \leq \rho \cdot \mathrm{OPT}(\tilde{d})$, and $\mathrm{OPT}(\tilde{d}) \leq (1+2\varepsilon)\mathrm{OPT}(d^*)$, $\sum_{j\in\mathcal{C}} w_j \tilde{d}(j,F) \leq \rho(1+2\varepsilon)\mathrm{OPT}(d^*)$. By Claims 5.1.3 and 5.1.4, $\sum_{j\in\mathcal{C}} w_j d^*(j,F) \leq (\rho(1+2\varepsilon)+\varepsilon)\mathrm{OPT}(d^*)$.

Query Complexity: Algorithm 3 uses queries to determine $S_{i,\ell}$ for all $i \in \mathcal{C}, \ell = 0,\ldots,q_i$. As we have the preference ranking for each agent, we have a list of agents sorted in non-decreasing order of their distance from i. Hence, to compute $S_{i,\ell}$, we can use binary search to determine maximal r_1, r_2 such that $r_1 < r_2$ and $d^*(i,\mathrm{alt}_\sigma(r_1)) \leq B_{i,0}(1+\varepsilon)^{-\ell} \leq d^*(i,\mathrm{alt}_\sigma(r_2))$. Then, $S_{i,\ell} = \{j \in \mathcal{C} : \mathrm{alt}_\sigma(r_1) \succeq_i j \succeq_i \mathrm{alt}_\sigma(r_2)\}$. The total number of value queries required to compute $S_{i,\ell}$ in this manner is $O(\log n)$, and hence the total number of value queries (per agent) that is needed to determine each of $S_{i,0},\ldots,S_{i,q_i}$ for a fixed agent i is $O(q_i \cdot \log n) = O(\log(n)\cdot\log(\alpha\rho\cdot n)/\varepsilon)$. □

As we will see in Section 5.3, we may not always be able to obtain a lower bound on the optimal value of weighted instance we are considering. In such settings, we can work with an upper bound on the optimal value, U, and still obtain the following modified guarantee.

Theorem 5.1.5. *Let $U \geq OPT(d^*)$ and $B \in [U, \alpha U]$. As before, let d^* be the true underlying metric, and let F be the set of centers opened by Algorithm 3. Then,*

$$\sum_{j\in\mathcal{C}} w_j d^*(j,F) \leq (\rho(1+2\varepsilon)+\varepsilon)U$$

Furthermore, Algorithm 3 can be implemented using $O(\log(n)\cdot\log(\alpha\rho\cdot n)/\varepsilon)$ value queries per agent.

5.2 Computing an estimate of OPT

In order to use the blackbox reduction to k-median (Algorithm 3), we must know B, a reasonable estimate of OPT. To compute such an estimate, we leverage the fact that OPT is at least the cost of a minimum-cost k-Forest, and is at most n times the cost of a minimum-cost k-Forest.

Claim 5.2.1. *Let OPT_{k-MCF} denote the cost of a minimum-cost k-Forest, and let $OPT_{k-median}$ denote the cost of an optimal k-median solution. Then,*

$$OPT_{k-MCF} \leq OPT_{k-median} \leq n \cdot OPT_{k-MCF}$$

Proof. Any k-median solution is a forest on k components (where the edges are between each agent and its assigned cluster center), so $\text{OPT}_{k-MCF} \leq \text{OPT}_{k-median}$. Let F^* be a minimum cost k-Forest. We can derive a k-median solution by choosing an arbitrary cluster center in each of the components induced by F^*, and assigning all clients in the cluster to this opened center. As we are preserving the components induced by F^*, due to the triangle inequality, the cost of this clustering is at most $n \cdot cost(F^*) = n \cdot \text{OPT}_{k-MCF}$. Thus, $\text{OPT}_{k-median} \leq \text{OPT}_{k-MCF}$. □

So, if we knew OPT_{k-MCF}, the value of a minimum-cost k-Forest, then $B = n \cdot \text{OPT}_{k-MCF}$ would satisfy $\text{OPT} \leq B \leq n \cdot \text{OPT}$.

If $d(i, j)$ was known for all $i, j \in \mathcal{C}$, an optimal minimum-cost k-Forest could be computed easily using Boruvka's algorithm. Boruvka's algorithm is a greedy minimum spanning tree (MST) algorithm, where at each stage, the cheapest edge incident to each (super)node is added and components are contracted into supernodes. The algorithm terminates when there is one supernode left. Given the MST, T, returned by Boruvka's algorithm (run with a fixed tie-breaking rule on the edges), we can remove the edges of T in non-increasing order of cost, until we obtain a forest with exactly k components; this is a minimum-cost k-Forest.

Of course, we do not know $d(i, j)$ for all $i, j \in \mathcal{C}$. Querying the value of $d(i, j)$ for all $i, j \in \mathcal{C}$ is computationally taxing on the agents, as this would take $\Omega(n)$ queries per agent. However, in order to run Boruvka's algorithm, we do not need to know the cost of *all* edges; we only need to know the minimum cost edge incident to each supernode. Hence, as we will show, only a few value queries are needed to run Boruvka's algorithm. The precise algorithm for computing the value of a minimum-cost k-Forest is given below.

Algorithm 4: Minimum cost k-Forest via Boruvka's algorithm

1 Fix a tie-breaking rule on the edges (that will be used in all subsequent edge-cost comparisons).
2 $F \leftarrow \varnothing$
3 $V_1 \leftarrow \mathcal{C}$
4 $E_1 \leftarrow \{\{i,j\} : i,j \in \mathcal{C}\}$
5 $t \leftarrow 1$
6 **while** $|V_t| > 1$ **do**
7 **for** $S \in V_t$ **do**
8 For each $v \in S$, query the value of $\min_{e \in \delta(v) \cap \delta(S)} d(e)$
9 Add $e = \arg\min_{e' \in \delta(S)} d(e')$ to F
10 **end**
11 Contract the components of $G_t = (V_t, F \cap E_t)$ into supernodes to get the (multi)graph $G_{t+1} = (V_{t+1}, E_{t+1})$
12 $t \leftarrow t + 1$
13 **end**
14 Sort F in non-increasing order of cost and remove edges in F until exactly k components are left
15 **return** $\sum_{e \in F} d(e)$

Lemma 5.2.2. *Algorithm 4 requires $O(\log n)$ queries per agent.*

Proof. Consider $S \in V_t$. For each $v \in S$, we know which edge attains $\min_{e \in \delta(v) \cap \delta(S)} d(e)$ (as we have the preference profile σ), so one value query is sufficient to compute the value of $\min_{e \in \delta(v) \cap \delta(S)} d(e)$. Given this, we can readily compute $e = \arg\min_{e' \in \delta(S)} d(e')$. Since each $v \in \mathcal{C}$ belongs to exactly one supernode of V_t, we incur the cost of one query per agent per iteration.

Since $|V_{t+1}| \leq \left\lceil \frac{|V_t|}{2} \right\rceil$, the while-loop (lines 6-13) terminates after $O(\log n)$ iterations; notice that the cost of every edge in F is known, so no additional value queries are needed in steps 14 and 15 of the algorithm. Thus, we make a total of $O(\log n)$ queries per agent. □

Combining this with the blackbox reduction of the previous section yields the following mechanism. Given the preference profile σ, compute B using Algorithm 4 and return the output of Algorithm 3 (when provided $\mathcal{C}, \sigma$, unit weights, B, and an $O(1)$-approximation algorithm for k-median as input). By Theorem 5.1.1, this is an $O(1)$-distortion mechanism that uses $O(\log^2(n))$ queries per agent.

5.3 Improving query complexity

For a constant $\varepsilon > 0$, the query complexity of using the $O(n)$-approximate estimate of OPT directly with Algorithm 3 is $O(\log^2 n)$; intuitively, this is due to the quality of our estimate of OPT, and the number of agents in our instance. Hence, in order to improve the query complexity, we should sparsify our instance before applying Algorithm 3. We can do this by using the bicriteria approximation algorithms for k-median that were described in Section 4.1.

Given a set S of $O(k)$ candidates, S induces a partition $\{P_j\}_{j \in S}$ of $\mathcal{C}$ (where P_j consists of all agents whose closest candidate is j, breaking ties in some arbitrary but consistent way). The weighted instance induced by S consists of the $O(k)$ agents in S and $\{w_j\}_{j \in S}$ where $w_j = |P_j|$. It is known that a good solution with respect to the weighted instance yields a good solution with respect to the original instance (see, for instance, the argument given by Charikar et. al [21])

Fact 5.3.1. Let $S \subseteq \mathcal{C}$ be a set such that $\sum_{j \in \mathcal{C}} d(j, S) \leq \alpha \cdot \text{OPT}$. Then, if OPT' is the optimal value of the weighted instance, $\text{OPT}' \leq 2(1 + \alpha)\text{OPT}$, and the solution obtained by running a ρ-approximation algorithm for k-median on the weighted instance induced by S has cost at most $(\alpha + 2\rho(1 + \alpha)) \cdot \text{OPT}$ with respect to the original instance.

Thus, if we obtain a constant factor bicriteria approximate solution for k-median using only a few value queries, we can apply the blackbox reduction to this sparsified instance. Unfortunately, we cannot use the cost of the bicriteria approximate solution as an estimate for OPT; since we may choose more than k candidates, the cost incurred by this approximate solution may in fact be strictly less than OPT. Nonetheless, running Algorithm

3 on the sparsified instance still yields a saving in the number of queries, even if we use $B \in [\mathrm{OPT}, n \cdot \mathrm{OPT}]$, as it has only $O(\log k)$ weighted points.

It remains to show how to obtain a bicriteria approximate solution, without using too many queries. Recall Meyerson's algorithm for online facility location (Algorithm 1), wherein a center is opened at x_i with probability $\min\{1, d(x_i, S)/f\}$ (S is the set of currently opened centers).

Corollary 4.1.3. *Let $x_1, \ldots, x_n$ be a sequence of clients in random order. Let $C_1^*, \ldots, C_k^*$ be the clusters induced by an optimal k-median solution. Let S be the set of centers opened by Algorithm 1 with $f = \frac{L}{k}$. Then, $\mathbb{E}\left[\sum_{j \in \mathcal{C}} d(j, S)\right] \leq 5L + 8 \cdot OPT$, and $\mathbb{E}\left[|S|\right] \leq \left(5 + 8 \cdot \frac{OPT}{L}\right) k$.*

Mechanism 5: $O(1)$-distortion, $O((\log(1/\delta)+\log k)\log n)$-query mechanism for k-median

Input: Preference profile σ

1 $B \leftarrow n \cdot \sum_{e\in T} d^*(e)$, where T is the min cost k-Forest returned by Algorithm 4
2 Obtain a random sequence of agents $x_1, \ldots, x_n$ via shuffling
3 $\mathcal{S} \leftarrow \varnothing$
4 **for** $i = 1, \ldots, \lceil \log_2 n \rceil + 1$ **do**
5 $L_i \leftarrow 2^{i-1} \cdot B/n$
6 $f \leftarrow \frac{L_i}{k}$
7 **repeat** $\log(1/\delta)$ **times**
8 $S \leftarrow \{x_1\}$
9 **for** $j = 2, \ldots, n$ **do**
10 Query x_j for the value of $d^*(x_j, top_S(x_j))$
11 Add x_j to S with probability $\min\{1, d^*(x_j, top_S(x_j))/f\}$
12 **end**
13 **if** $|S| \leq 52k$ **then**
14 $\mathcal{S} \leftarrow \mathcal{S} \cup \{S\}$
15 **end**
16 **end**
17 **end**
18 Choose $S \in \arg\min_{S'\in\mathcal{S}} \sum_{j\in\mathcal{C}} d^*(j, S')$; $\{P_j\}_{j\in S}$ is the partition induced by S
19 For $j \in S$, $w_j \leftarrow |P_j|$
20 $\overline{S} \leftarrow$ Algorithm3$(S, \sigma, \{w_j\}_{j\in S}, B, \mathcal{A})$ where $\mathcal{A}$ is an $O(1)$-approximation algorithm for k-median.
21 **return** $\overline{S}$

In step 10, we compute $\delta_i = d(x_i, S)$ by querying $d(x_i, top_S(x_i))$, which is the distance between x_i, and the candidate in S that is closest to x_i.

Theorem 5.3.2. *Mechanism 5 is an $O(1)$-distortion mechanism for k-median that requires $O((\log(1/\delta)+\log k)\log n)$ value queries per agent, and has a success probability of at least*

$1 - \delta$.

Proof. Let $\varepsilon \in (0, 1]$ be a constant. Notice that in lines 7-16, we are running Meyerson's algorithm (Algorithm 1) $\log(1/\delta)$ times for a given L_i (which serves as our estimate for OPT). Moreover, since we know that $\text{OPT} \leq B \leq n \cdot \text{OPT}$, there exists some $i^* \in \{1, \ldots, \lceil \log_2 n \rceil\}$ such that $\text{OPT} \leq L_{i^*} \leq 2 \cdot \text{OPT}$.

It suffices to show that the with probability at least $1 - \delta$, one of the solutions returned by Meyerson's algorithm when $f = L_{i^*}/k$ opens at most $52k$ centers, and induces a total connection cost of at most $72\text{OPT}(d^*)$. By Corollary 4.1.3 and Markov's inequality, the connection cost induced by the output of Meyerson's algorithm when $f = L_{i^*}/k$ is at most $4 \cdot 18\text{OPT}$ with a probability of $\frac{3}{4}$; and the number of centers opened is at most $4 \cdot 13k = 52k$, with a probability of $\frac{3}{4}$. Hence, by Union bound, with a probability of at least $\frac{1}{2}$, the solution returned by Meyerson's algorithm is a (72, 52)-bicriteria approximate k-median solution. Since we run Meyerson's algorithm $\log(1/\delta)$ times, with probability at least $1 - \delta$, there exists $S \in \mathcal{S}$ such that $\sum_{j \in \mathcal{C}} d^*(j, S) \leq 72 \cdot \text{OPT}$ and $|S| \leq 52k$. As noted before, this S yields a sparsified (weighted) instance.

We would like to apply the blackbox reduction (Algorithm 3) to this sparsified instance. However, it is possible that OPT', the optimal value of the weighted instance is less than OPT, and hence $\text{OPT}' \leq B \leq n \cdot \text{OPT}'$ may not hold. By Fact 5.3.1, $\text{OPT}' \leq 2(72+1)\text{OPT}$; if we take $U = 146\text{OPT}$, $U \leq 146B \leq n \cdot U$, and hence we can apply Theorem 5.1.5. This yields a $146(\rho(1 + 2\varepsilon) + \varepsilon)$-approximate solution to the original instance, where ρ is the approximation factor of the k-median algorithm used in step 20. In particular, if we use Byrka et. al's $(2.675 + \varepsilon)$-approximation algorithm (which has the current-best approximation factor for k-median), we obtain a solution of cost at most $146((2.675 + \varepsilon)(1 + 2\varepsilon) + \varepsilon)\text{OPT}(d^*)$.

Query Complexity: The steps which require value queries are lines 4-18, and the call to Algorithm 3 in line 20. The total number of queries made in lines 4-17 is $O(\log(1/\delta) \cdot \log(n))$. Since $|\mathcal{S}| = O(\log(1/\delta) \cdot \log n)$, the number of queries per agent to find $S \in \arg\min_{S' \in \mathcal{S}} \sum_{j \in \mathcal{C}} d^*(j, S')$ is at most $O(\log(1/\delta) \cdot \log n)$ queries per agent. Finally, since the weighted instance given as input to Algorithm 3 in line 20 consists of $O(k)$ points and

$B \in [\text{OPT}, n \cdot \text{OPT}]$, this step takes at most $O(\log n \log k/\varepsilon)$ queries per agent (by Theorem 5.1.1). □

5.4 Query complexity independent of n

In some applications, the number of agents can be much larger than k; hence, a natural question to ask is whether there exists a mechanism whose query complexity is independent of n. We can leverage Aggarwal et. al's adaptive sampling algorithm (Algorithm 2), which does not require an estimate of OPT in order to compute a bicriteria solution. Recall that this algorithm successively chooses the candidate with probability proportional to the distance from the set of currently chosen candidates. For a given agent j and set of candidates S, the distance from j to S is $d(j, top_S(j))$, which can be computed using one value query.

Mechanism 6: $O(1)$-distortion, $O(\log(1/\delta)k)$-query mechanism for k-median

Input: Preference profile σ

1 $\mathcal{S} \leftarrow \varnothing$
2 **repeat** $\log(1/\delta)$ **times**
3 $S \leftarrow \varnothing$
4 **for** $i = 1, \ldots, 16(k + \sqrt{k})$ **do**
5 **for** $j \in \mathcal{C}$ **do**
6 Query j for the value of $d^*(j, top_S(j))$
7 **end**
8 Sample s_i with probability proportional to $d^*(s_i, S)$
9 Update $S \leftarrow S \cup \{s_i\}$
10 **end**
11 $\mathcal{S} \leftarrow \mathcal{S} \cup \{S\}$
12 **end**
13 Choose $S \in \arg\min_{S' \in \mathcal{S}} \sum_{j \in \mathcal{C}} d(j, S')$; $\{P_j\}_{j \in S}$ is the partition induced by S
14 **for** $j \in S$ **do**
15 Query $d^*(i, j)$ for all $i \in S \setminus \{j\}$
16 **end**
17 $\bar{S}$: output of $(3 + \varepsilon)$-approximation algorithm for k-median on the (cardinal) weighted instance (S, w, d) where $w_j = |P_j|$ for all $j \in S$
18 **return** $\overline{S}$

Theorem 5.4.1. *Mechanism 6 is an $O(1)$-distortion mechanism for k-median that requires $O(\log(1/\delta)k)$ value queries per agent, and has a success probability of at least $1 - \delta$.*

Proof. Notice that in lines 2-12, we are running Aggarwal, Deshpande, and Kannan's adaptive sampling algorithm (Algorithm 2) $\log(1/\delta)$ times. So, with probability at least $1 - \delta$, by Theorem 4.1.4, there exists $S \in \mathcal{S}$ such that $\sum_{j \in \mathcal{C}} d^*(j, S) \leq 20 \cdot \text{OPT}$. Finally, given this $(20, 16(1 + \frac{1}{\sqrt{k}}))$-bicriteria approximate k-median solution, we query all pairwise distances for $i, j \in S$ and obtain $\overline{S}$ by solving the weighted instance induced by S using a $(3 + \varepsilon)$-approximation algorithm for k-median; by Fact 5.3.1, $\sum_{j \in \mathcal{C}} d^*(j, \overline{S}) \leq (20 + 42(3 +$

$\varepsilon)) \times \mathrm{OPT}$.

Query Complexity: The steps which require value queries are lines 2-12, and lines 14-16. The total number of value queries made in lines 14-16 is $O(k)$ queries per agent in S (since $|S| = O(k)$). Since the inner loop in lines 2-12 makes $16(k + \sqrt{k}) + 1$ queries per agent, the total number of queries per agent that are made in lines 2-12 is $O(\log(1/\delta) \cdot k)$. Thus, the total number of queries per agent is $O((\log(1/\delta) + 1) \cdot k)$. □

Chapter 6

Beyond social cost minimization: $O(1)$-distortion algorithms for the ℓ-centrum problem

Until now, we have exclusively considered social cost minimization problems where one seeks to minimize the *total* cost incurred by the agents. However, depending on the application, this utilitarian objective may not be the appropriate choice. For instance, in some settings where fairness is important, we may wish to consider an egalitarian objective and minimize the *maximum* cost incurred by any agent. The utilitarian and egalitarian objectives are special cases of the Top_ℓ objective, wherein one wishes to minimize the sum of the ℓ largest costs incurred by the agents. When $\ell = 1$ and $\ell = n$ we recover the egalitarian and utilitarian objectives respectively.

In this chapter, we study the problem of electing a committee of k candidates that minimizes the sum of the ℓ largest costs incurred by the agents. We will refer to ℓ-centrum and k-winner selection under the Top_ℓ objective interchangeably throughout the rest of this thesis. When $k = 1$, there exists a social choice function that has a distortion of at most 3 with respect to the Top_ℓ objective [24]. For $k > 3$, the example in Chapter 4 shows that distortion of any social choice correspondence is unbounded; hence, as with k-median, we will focus our attention on designing a mechanism that uses a limited number of value

queries. As before, let $\mathcal{M} = (\mathcal{Q}, f, k)$ be a mechanism. In the Top_ℓ setting, we define the distortion of $\mathcal{M}$ to be

$$\mathrm{distortion}(\mathcal{M}) = \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\mathrm{Top}_\ell(d(\mathcal{C}, \mathcal{M}(\sigma|d)))}{\min_{S \subseteq \mathcal{C}^k} \mathrm{Top}_\ell(d(\mathcal{C}, S))}$$

where $d \triangleleft \sigma$ denotes that d is consistent with σ.

We adapt the ideas used to design the k-median mechanisms presented in Chapter 5 to obtain $O(1)$-distortion mechanisms for the ℓ-centrum problem with analogous query complexities. Specifically, we show the following.

- One of the insights in Chapter 5 was that if we could approximate the true metric d^* with a simulated metric, then we can run any approximation algorithm for the cardinal setting on the simulated metric. This idea applies to ℓ-centrum as well. In Section 6.1, we design a blackbox reduction to the cardinal setting (see Algorithm 7), by suitably adapting the reduction in Section 5.1 (Algorithm 3) for the k-median problem. Utilizing this reduction directly in conjunction with an $O(1)$-approximation algorithm for ℓ-centrum (in the cardinal setting) yields an $O(1)$-distortion mechanism with $O(\log^2 n)$ per-agent query complexity (Theorem 6.2.4).

- In Section 6.2, we obtain an improved query complexity of $O(\mathit{polylog}(k) \cdot \log(n))$. The key idea here is to show that Meyerson's algorithm, which yields a constant-factor bicriteria solution for k-median (given a suitable estimate of OPT), can be adapted to obtain a similar guarantee for ℓ-centrum. As before, running the blackbox reduction followed by an ℓ-centrum algorithm on the sparsified weighted instance obtained by consolidating points at the centers output by Meyerson's algorithm then yields the improved query complexity.

- In Section 6.3 we devise a mechanism with query complexity independent of n. First, we show that the adaptive-sampling approach for k-median (Algorithm 2) can be leveraged in a novel fashion to obtain a constant-factor bicriteria approximation for ℓ-centrum, given a suitable guess of a certain statistic of the optimal solution. Running an ℓ-centrum approximation algorithm on the weighted instance resulting from this bicriteria solution then yields an $O(1)$-distortion mechanism with $O(k \log \ell)$ queries per agent.

In the context of the ℓ-centrum problem, a *weighted instance* consists of a set of agents, $\mathcal{C}$, with non-negative weights $\{w_i\}_{i\in\mathcal{C}}$. Each weighted point i with integer weight $w_i \geq 0$ denotes w_i co-located points. Then, the cost vector induced by a solution T, which we denote as $d(\mathcal{C}, T|w)$, is a vector in R^W, where $W = \sum_i w_i$, and the Top_ℓ cost is the Top_ℓ-norm of this vector. For a set $S \subseteq \mathcal{C}$, the *weighted instance induced by* S is $(\mathcal{C}, \{w_i\}_{i\in\mathcal{C}})$ where $w_i = 0$ if $i \notin S$, and $w_j = |P_j|$ otherwise (recall that P_j are the agents in $\mathcal{C}$ that have been assigned to $j \in S$).

In the k-median setting, we heavily utilized Fact 5.3.1, which asserts that we will not incur a large cost when working with the sparsified instance, instead of the original instance. We will prove that an analogous statement holds for the ℓ-centrum problem as well.

Lemma 6.0.1. *Let $S \subseteq \mathcal{C}$ such that the Top_ℓ value of $d(\mathcal{C}, S)$, the assignment cost vector induced by S, is at most $\alpha \cdot OPT$. Let OPT' be the optimal value for the ℓ-centrum problem on the weighted instance induced by S, and OPT be the optimal value of the original instance. Then,*

1. *$OPT' \leq 2(\alpha + 1)\,OPT$*
2. *If T is a ρ-approximate solution with respect to the weighted instance, $\text{Top}_\ell(d(\mathcal{C}, T)) \leq (\alpha + 2\rho(\alpha + 1)) \cdot OPT$*

Proof. We first prove that (a) holds. Let T^* be an optimal solution for the original instance, and denote the optimal value of the original instance as OPT. Let $\tilde{T}$ be the projection of T^* onto S, that is, the centers obtained by mapping each point in T^* to the closest center in S. We show an upper bound on $\text{Top}_\ell(d(\mathcal{C}, \tilde{T}|w))$, the Top_ℓ-cost of the weighted instance with respect to $\tilde{T}$. Consider any subset of ℓ points, Q (where we take the weights into consideration, i.e., we take some w'_i points from each $i \in S$, where $\sum_{i\in S} w'_i = \ell$).

For each $i \in Q$, let $x_S(i)$ be the point that i is co-located with in the weighted instance, and $x^*(i)$ be the center in T^* that is closest to i. By the triangle inequality,

$$\sum_{i\in Q} d(x_S(i), \tilde{T}) \leq \sum_{i\in Q} d(x_S(i), i) + \sum_{i\in Q} d(i, x^*(i)) + \sum_{i\in Q} d(x^*(i), \tilde{T})$$

The first term, $\sum_{i \in Q} d(x_S(i), i)$, is the cost incurred when we move each $i \in Q$ from $x_S(i)$ to its original location; this is at most $\text{Top}_\ell(d(\mathcal{C}, S))$. The second term, $\sum_{i \in Q} d(i, x^*(i))$, is the cost of moving each $i \in Q$ from its original location to $x^*(i)$, its closest center in T^*; the cost of this step is at most OPT. Finally, $\sum_{i \in Q} d(x^*(i), \tilde{T})$ is the cost of moving the points their centers in T^* to their closest open centers in $\tilde{T}$. The cost of this step can be bounded by moving each relevant point in T^* to $\tilde{T}$ – so we incur an additional cost of at most $\text{OPT} + \text{Top}_\ell(d(\mathcal{C}, S))$. Putting this together, we have

$$\sum_{i \in Q} d(x_S(i), \tilde{T}) \leq \sum_{i \in Q} d(x_S(i), i) + \sum_{i \in Q} d(i, x^*(i)) + \sum_{i \in Q} d(x^*(i), \tilde{T}) \leq 2\text{Top}_\ell(d(\mathcal{C}, S)) + 2\text{OPT}$$

As this holds for any ℓ-subset Q, $\text{Top}_\ell(d(\mathcal{C}, \tilde{T})) \leq 2(\text{OPT} + \text{Top}_\ell(d(\mathcal{C}, S))) \leq 2(\alpha + 1)\text{OPT}$.

It remains to prove that (b) holds. For any solution, T, of Top_ℓ cost Z for the weighted instance, the cost of T for the original instance is at most $Z + \text{Top}_\ell(d(\mathcal{C}, S))$ (this is an upper bound on the cost of moving the ℓ weighted points to their original locations). Since $\text{OPT}' \leq 2(\alpha + 1)\text{OPT}$, for any ρ-approximate solution T for the weighted instance, $\text{Top}_\ell(d(\mathcal{C}, T)) \leq (\alpha + 2\rho(\alpha + 1))\text{OPT}$.

□

6.1 A blackbox reduction for the Top_ℓ setting

In the k-median setting, we observed that, if the true underlying metric is (approximately) known, we can leverage existing (cardinal) k-median algorithms. Given a reasonable estimate of OPT, Algorithm 3 allowed us to construct a *simulated metric* that, in a sense, approximated the true underlying metric. With a few modifications, we can extend this blackbox reduction to the Top_ℓ setting.

The key difference between Algorithm 7 and the blackbox reduction for k-median (Algorithm 3) is that instead of working with the provided *positive* weight w_i, we use $w'_i = \min(w_i, \ell)$; this ensures that at most ℓ agents co-located at i can contribute to the Top_ℓ-cost.

Algorithm 7: A blackbox reduction to the ℓ-centrum problem

Input: A set of n agents $\mathcal{C}$ with preference profile σ, and non-negative, integer weights $\{w_j\}_{j\in\mathcal{C}}$; d^* denotes the true underlying metric (which we do not know).

A constant B such that $B \in [\mathrm{OPT}(d^*), \alpha \cdot \mathrm{OPT}(d^*)]$

A ρ-approximation algorithm $\mathcal{A}$ for solving ℓ-centrum on the weighted instance

for $i \in \mathcal{C}$ **do**
  Define $w'_i = \min(w_i, \ell)$, $B_{i,0} = \rho(1+3\varepsilon) \cdot \frac{B}{w'_i}$, $q_i := \left\lceil \log_{1+\varepsilon}(\frac{\alpha w'_i B_{i,0} \cdot \ell}{\varepsilon B}) \right\rceil$
  Compute $S_{i,r} = \{j \in \mathcal{C} : d^*(i,j) \leq B_{i,0}(1+\varepsilon)^{-\ell}\}$ for $r = 0, \ldots, q_i$
end
Compute $\tilde{d}$ such that $\tilde{d}$ is a valid metric and also satisfies the following constraints:

(1) $\tilde{d}(i,j) \geq B_{i0}$ for all $j \notin S_{i,0}$.

(2) $(1+\varepsilon)^{-(r+1)} B_{i0} \leq \tilde{d}(i,j) \leq (1+\varepsilon)^{-r} B_{i0}$ for all $j \in S_{i,r} \setminus S_{i,r+1}$, $i \in \mathcal{C}, r \in \{0, \ldots, q_i - 1\}$

(3) $\tilde{d}(i,j) \leq \frac{\varepsilon B}{\alpha \ell \cdot w'_i}$ for all $j \in S_{i,q_i}$

return $\mathcal{A}(\mathcal{C}, w', \tilde{d})$

Theorem 6.1.1. *Let d^* be the true underlying metric, and let F be the set of centers opened by Algorithm 7. Then,*

$$Top_\ell(d^*(j, F|w)) \leq (\rho(1+2\varepsilon) + \varepsilon) OPT(d^*)$$

Furthermore, Algorithm 7 can be implemented using $O(\log(n) \cdot \log(\alpha\rho \cdot n)/\varepsilon)$ value queries per agent.

Proof. The proof of this theorem is analogous to the proof of Theorem 5.1.1; for the sake of brevity, we will emphasize the differences and omit the redundant details. Let $\mathrm{OPT}(\tilde{d})$

denote the value of an optimal ℓ-centrum solution computed with respect to $\tilde{d}$. Fact 5.1.2 still holds (if we substitute w_i with w'_i and define $\kappa_i = \frac{\varepsilon B}{\alpha \ell w'_i}$). Given this, we can show the following claim

Claim 6.1.2. *Let $T \subseteq \mathcal{C}$ such that $d^*(i,T) \leq B_{i,0}$ for all $i \in \mathcal{C}$. Then,*

(a) $Top_\ell(\tilde{d}(\mathcal{C},T|w)) \leq (1+\varepsilon)\, Top_\ell(d^*(\mathcal{C},T|w)) + \varepsilon\, OPT(d^*)$

(b) $Top_\ell(d^*(\mathcal{C},T|w)) \leq Top_\ell(\tilde{d}(\mathcal{C},T|w)) + \varepsilon\, OPT(d^*)$

Proof. Let Q be a set of ℓ agents (where we take the weights into consideration, i.e., we take some w'_i points from each $i \in S$, where $\sum_{i \in S} w'_i = \ell$). Since $d^*(i,T) \leq B_{i,0}$ for all $i \in \mathcal{C}$, by Fact 5.1.2,

$$\sum_{i \in Q} d^*(i,T) \leq (1+\varepsilon) \sum_{i \in Q} \tilde{d}(i,T) + \sum_{i \in Q} w'_i \kappa_i \leq (1+\varepsilon) \mathrm{Top}_\ell(\tilde{d}(\mathcal{C},T|w)) + \sum_{i \in Q} w'_i \kappa_i$$

Since $|Q| = \ell$, $\sum_{i \in Q} w'_i \kappa_i \leq \ell \cdot \frac{\varepsilon \mathrm{OPT}(d^*)}{\ell}$. As this holds for any ℓ-subset Q,

$$\mathrm{Top}_\ell(\tilde{d}(\mathcal{C},T|w)) \leq (1+\varepsilon) \mathrm{Top}_\ell(d^*(\mathcal{C},T|w)) + \varepsilon \mathrm{OPT}(d^*)$$

The proof of (b) is identical, and hence omitted. □

Moreover, by Claim 6.1.2(a), $\mathrm{OPT}(\tilde{d}) \leq (1+2\varepsilon) \cdot \mathrm{OPT}(d^*)$. We now show that $d^*(j,T) \leq B_{i,0}$ for all $j \in \mathcal{C}$, for any ρ-approximate solution (with respect to $\tilde{d}$).

Claim 6.1.3. *Let $T \subseteq \mathcal{C}$. If $Top_\ell(\tilde{d}(\mathcal{C},T|w)) \leq \rho \cdot OPT(\tilde{d})$, $d^*(i,T) \leq B_{i,0}$ for all $i \in \mathcal{C}$.*

Proof. Suppose, to arrive at a contradiction, that there exists $j \in \mathcal{C}$ such that $d^*(j,T) > B_{j,0}$. Since $w'_j \leq \ell$, at least w'_j agents who contribute to the Top_ℓ objective incur a connection cost of $d(j,T)$ or larger, so,

$$\mathrm{Top}_\ell(\tilde{d}(\mathcal{C},T|w))| \geq w'_j \cdot \tilde{d}(j,T) \geq w'_j B_{j,0} > \rho(1+2\varepsilon)\mathrm{OPT}(d^*) \geq \rho \cdot \mathrm{OPT}(\tilde{d})$$

which is a contradiction. □

Let F^* be the set of centers opened by an optimal solution with respect to d^* and let F be the set of centers opened by $\mathcal{A}(\mathcal{C}, w', \tilde{d})$. By Claims 6.1.2 and 6.1.3, $\text{Top}_\ell(d^*(\mathcal{C}, F|w)) \leq (\rho(1+2\varepsilon)+\varepsilon)\text{OPT}(d^*)$.

Query Complexity: As done in Algorithm 3, we can combine binary search with queries $S_{i,r}$ for all $i \in \mathcal{C}, r = 0, \ldots, q_i$. The total number of value queries required to compute $S_{i,r}$ in this manner is $O(\log n)$, and hence the total number of value queries (per agent) that is needed to determine each of $S_{i,0}, \ldots, S_{i,q_i}$ for a fixed agent i is $O(q_i \cdot \log n) = O(\log(n) \cdot \log(\alpha\rho\ell n)/\varepsilon)$. □

One can show that following guarantee also holds for Algorithm 7; the proof of this proposition is almost identical to the proof of Theorem 6.1.1, and hence omitted.

Theorem 6.1.4. *Let $U \geq OPT(d^*)$ and $B \in [U, \alpha U]$. As before, let d^* be the true underlying metric, and let F be the set of centers opened by Algorithm 7. Then,*

$$Top_\ell(d^*(j, F|w)) \leq (\rho(1+2\varepsilon)+\varepsilon)U$$

Furthermore, Algorithm 7 can be implemented using $O(\log(n) \cdot \log(\alpha\rho \cdot n)/\varepsilon)$ value queries per agent.

In order to use this blackbox reduction to ℓ-centrum, we must know B, a reasonable estimate of OPT. To compute such an estimate, we leverage the fact that the cost of an optimal k-median solution is at least the cost of the optimal ℓ-centrum solution, and at most n times the cost of the optimal ℓ-centrum solution. In Chapter 5, we saw that we can compute OPT_{k-MCF} using $O(\log n)$ queries per agent (using Algorithm 4); by this argument, $\text{OPT}_{\ell\text{-centrum}} \leq n \cdot \text{OPT}_{k-MCF} \leq n^2 \cdot \text{OPT}_{\ell\text{-centrum}}$. Combining this with Algorithm 7 yields the following mechanism. Given the preference profile σ, compute OPT_{k-MCF} using Algorithm 4 and return the committee selected by Algorithm 7 (when provided $\mathcal{C}, \sigma$, unit weights, $B = n \cdot \text{OPT}_{k-MCF}$, and an $O(1)$-approximation algorithm for ℓ-centrum as input). By Theorem 6.1.1, this is an $O(1)$-distortion mechanism that uses $O(\log^2(n)/\varepsilon)$ queries per agent.

6.2 Improving query complexity

Once again, in order to improve the query complexity of our mechanism, we will sparsify our instance using a bicriteria approximation algorithm. In Chapter 5, we showed that Meyerson's algorithm for online facility location and Aggarwal et. al's adaptive sampling algorithms could be used to sparsify the instance (using relatively few value queries per agent). In this section, we show that Meyerson's algorithm for online facility location can be extended to solve the ℓ-centrum k-clustering problem as well. This allows us to devise a mechanism that is analogous to Mechanism 5 for the Top_ℓ setting.

Algorithm 8: Extension of Meyerson's OFL algorithm for ℓ-centrum k-clustering

Input: Sequence of agents $x_1, \ldots, x_n$,
A constant B such that $\text{OPT} \leq B \leq \alpha \cdot \text{OPT}$

$S \leftarrow \{x_1\}$
$f = \frac{B}{k}$
for $i = 2, \ldots, n$ **do**
  $\delta_i = \left(d(x_i, S) - 3 \cdot \frac{B}{\ell}\right)^+$
  Add x_i to S with probability $\min(1, \delta_i / f)$
end
return S

Theorem 6.2.1. *Let OPT be the optimal value of an ℓ-centrum k-clustering on $\mathcal{C}$. If the order of agents is random, the expected number of facilities opened by Algorithm 8 is at most $26k$, and the expected cost is at most $15B + 14OPT$.*

Proof of Theorem 6.2.1 (Adapted from [35]). The key to adapting Meyerson's algorithm to the Top_ℓ-setting is the fact that, as stated in Chapter 2, the Top_ℓ cost of a vector $v \in \mathbb{R}^n$ can be well-approximated by the *separable* proxy function $\ell \cdot t + \sum_{i=1}^n (v_i - t)^+$, under a suitable choice of t. Given an estimate B of the optimum value, one can use $t = B/\ell$, and therefore focus on the second term $\sum_{i=1}^n (v_i - t)^+$. Roughly speaking, we can then treat $(d(j, S) - t)^+$ as the connection cost when running Meyerson (where S is the set of currently open centers). However, certain complications arise since this does not

quite satisfy the triangle inequality, and therefore we actually work with connection-cost expression $(d(j,S)-3t)^+$. Our analysis will bound the expected value of this expression, which then also yields a bound on the expected Top$_\ell$-cost (via Claim 2.2.1)

Let S^* be the centers opened by an optimal ℓ-centrum solution, and let $C_1^*,\ldots,C_k^*$ be the clusters induced by S^*. Suppose that S be the set of centers opened by Algorithm 8. For an agent $p\in\mathcal{C}$, $S_p\subseteq S$ is the set of currently open centers when p is considered, and hence $\delta_p=(d(p,S_p)-3\cdot\frac{B}{\ell})^+$. For ease of exposition, we will define $t_\ell:=\frac{B}{\ell}$.

Consider cluster C_i^*, with cluster center c_i^*. For $i=1\ldots,k$, we define the *average ℓ-radius* of C_i^* to be $r_\ell(C_i^*)=\sum_{j\in C_i^*}\frac{(d(j,c^*)-t_\ell)^+}{|C^*|}$. The *$\ell$-core* of C_i^* is

$$core_\ell(C_i^*)=\{j\in C_i^*:(d(j,c^*)-t_\ell)^+\leq 2r_\ell(C_i^*)\}$$

We follow Meyerson's approach [35] and bound the expected cost incurred by the agents in $core_\ell(C_i^*)$ and not in $core_\ell(C_i^*)$ separately. To be precise, we bound $\mathbb{E}[\min(\delta_g,f)]$ for all $g\in core_\ell(C_i^*)$, and then bound $\mathbb{E}[\min(\delta_b,f)]$ for agents $b\in C^*\setminus core_\ell(C_i^*)$ in terms of the cost of the last core-agent preceding b.

We begin by bounding $\sum_{j\in core_\ell(C_i^*)}\mathbb{E}[\min(\delta_j,f)]$. Once a center $g_{j^*}\in core_\ell(C_i^*)$ has been opened, we have $\delta_j\leq(d(j,c')-3t_\ell)^+\leq(d(j,c_i^*)-2t_\ell)^+ + (d(c',c_i^*)-t_\ell)^+\leq(d(j,c_i^*)-t_\ell)^+ + 2r_\ell(C_i^*)$, for every subsequent $j\in C_i^*$. It remains to bound $\mathbb{E}[\min(\delta_g,f)]$ for core-agents g that precede g_{j^*}. To this end, we use the following lemma proved by Liberty et. al [33]

Lemma 6.2.2 (Lemma 2.1 in [33]). *We are given a sequence $X_1,\ldots X_n$ of n independent experiments. Each experiment succeeds with probability $p_i\geq\min\{A_i/B,1\}$ where $B\geq 0$ and $A_i\geq 0$ for all $i=1,\ldots,n$. Let t be the (random) number of consecutive unsuccessful experiments before the first successful one, then:*

$$\mathbb{E}\left[\sum_{i=1}^{t}A_i\right]\leq B$$

The events of opening centers at core-agents are independent when we condition on the sequence in which core-agents are considered, the centers opened outside the core, and the number of core-agents considered before a center is opened for the first time. To be precise,

let $q = |core_\ell(C_i^*)|$ (so $q \geq \frac{|C_i^*|}{2}$) and let $g_1, \ldots, g_q$ be the order in which the core-agents in $core_\ell(C_i^*)$ are considered. Let g_{j^*} be the first core-agent at which a center is opened. For $i = 1, \ldots, j^* - 1$, the probability of opening a center at g_i is $\frac{\min(\delta_{g_i}, f)}{f}$; hence, by Lemma 6.2.2,

$$\sum_{i=1}^{j^*-1} \mathbb{E}\left[\min(\delta_{g_i}, f)\right] \leq f$$

where the expectation is conditioned on the above events. Since $\min(\delta_{g_{j^*}}, f) \leq f$, we have

$$\sum_{g \in core_\ell(C_i^*)} \mathbb{E}[\min(\delta_g, f)] \leq 2f + \sum_{g \in core_\ell(C_i^*)} \left((d(g, c_i^*) - t_\ell)^+ + 2r_\ell(C_i^*)\right) \tag{6.1}$$

We now bound $\mathbb{E}[\min(\delta_b, f)]$ for an agent $b \in C_i^* \setminus core_\ell(C_i^*)$. First, if b precedes all core agents, we simply bound $\min(\delta_b, f)$ by f. Note that this case happens with probability $\frac{1}{q+1} \leq \frac{2}{|C_i^*|}$.

Suppose b is preceded by some core-agent g. Let S_g be the set of centers that are open immediately after g is considered. By the triangle inequality, $\delta_b \leq (d(b, S_g) - 3t_\ell)^+ \leq (d(b, c_i^*) - t_\ell)^+ + (d(g, c_i^*) - t_\ell)^+ + (d(g, S_g) - t_\ell)^+$. Moreover, as $g \in core_\ell(C_i^*)$, $(d(g, c_i^*) - t_\ell)^+ \leq 2r_\ell(C_i^*)$. We consider two cases here:

- If $d(g, S_g) > 4t_\ell$, $(d(g, S_g) - t_\ell)^+ \leq 3(d(g, S_g) - 3t_\ell)^+$. Combining this with the earlier bound on δ_b yields

  $$\delta_b \leq (d(b, c_i^*) - t_\ell)^+ + 2r_\ell(C_i^*) + 3(d(g, S_g) - 3t_\ell)^+ = (d(b, c_i^*) - t_\ell)^+ + 2r_\ell(C_i^*) + 3\delta_g$$

  Since no center is open at g, $\min\{\delta_g, f\} = \delta_g$. Thus, $\delta_b \leq (d(b, c_i^*) - t_\ell)^+ + 2r_\ell(C_i^*) + 3\min(\delta_g, f)$.

- If $d(g, S_g) \leq 4t_\ell$, $\delta_b \leq (d(b, c_i^*) - t_\ell)^+ + 2r_\ell(C_i^*) + 3t_\ell$.

We now utilize these to bound $\mathbb{E}[\min(\delta_b, f)]$ by conditioning on the order in which core-agents appear. Let $g_1, \ldots, g_q$ be this ordering. We also condition on the first core-agent (if one exists) for which $d(g, S_g) \leq 4t_\ell$; let g_t denote this agent. Let $prev(b)$ denote the last core-agent that precedes b (if no such agent exists, $prev(b) = \varnothing$).

Claim 6.2.3. *Conditioned on the above events, we have* $\Pr[prev(b) = \varnothing] = \frac{1}{q+1}$*, and for any* $g \in core_\ell(C_i^*)$*,* $\Pr[prev(b) = g] = \frac{1}{q+1}$.

Proof. This follows due to the ordering of the agents; because the order of the agents is random, each candidate for $prev(b)$ in $\{\varnothing\} \cup core_\ell(C_i^*)$ is equally likely. □

So, conditioned on the above events, we have

$$\begin{aligned}\mathbb{E}[\min(\delta_b, f)] &\leq \Pr[prev(b) = \varnothing] \cdot f + (d(b, c_i^*) - t_\ell)^+ + 2r_\ell(C_i^*) \\ &\quad + \sum_{i=1}^{t-1} \Pr[prev(b) = g_i] 3\mathbb{E}[\min(\delta_{g_i}, f)] + \sum_{i=t}^{q} \Pr[prev(b) = g_i] \cdot 3t_\ell \\ &\leq \frac{2f}{|C^*|} + (d(b, c_i^*) - t_\ell)^+ + 2r_\ell(C_i^*) + \frac{2}{|C_i^*|} \sum_{g \in core_\ell(C_i^*)} 3\mathbb{E}[\min(\delta_g, f)] \\ &\quad + \frac{2}{|C_i^*|} \cdot |C_i^* \setminus core_\ell(C_i^*)| \cdot 3t_\ell\end{aligned}$$

(where the expectation in the final inequality is again conditioned on the above events). Summing over all non-core agents gives the following. Note that $|C_i^* \setminus core_\ell(C_i^*)| \leq \frac{|C_i^*|}{2}$. We have

$$\begin{aligned}\sum_{b \in C_i^* \setminus core_\ell(C_i^*)} \mathbb{E}[\min(\delta_b, f)] \leq f &+ \sum_{b \in C_i^* \setminus core_\ell(C_i^*)} (d(b, c_i^*) - t_\ell)^+ + |C_i^* \setminus core_\ell(C^*)| \cdot 2r_\ell(C_i^*) \\ &+ \quad 3 \sum_{g \in core_\ell(C_i^*)} \mathbb{E}[\min(\delta_g, f)] + 3|C_i^* \setminus core_\ell(C_i^*)| \cdot t_\ell\end{aligned}$$

Summing over all core and non-core agents, and over all clusters $C_1^*, \ldots, C_k^*$ yields

$$\begin{aligned}&\sum_{j \in \mathcal{C}} \mathbb{E}[\min(\delta_j, f)] \\ &\leq 9kf + \sum_{i=1}^{k} 4 \sum_{p \in C_i^*} (d(p, c_i^*) - t_\ell)^+ + 10|C_i^*| \cdot r_\ell(C_i^*) + 3|C_i^* \setminus core_\ell(C_i^*)| \cdot t_\ell\end{aligned}$$

Recall that S^* is the set of centers opened by an optimal solution and $t_\ell = \frac{B}{\ell}$. Since $\text{OPT} \leq B \leq \alpha \cdot \text{OPT}$, the ℓth largest assignment cost induced by S^* is at most $\frac{B}{\ell}$, and hence $|\{j \in \mathcal{C} : d(j, S^*) > t_\ell\}| \leq \ell$. So, $\sum_{i=1}^{k} |C_i^* \setminus core_\ell(C_i^*)| \leq \ell$. Furthermore, $\sum_{j \in \mathcal{C}} (d(j, S^*) - t_\ell)^+ \leq \text{OPT}$ and for any $i \in [k]$, $|C_i^*| \cdot r_\ell(C_i^*) = \sum_{j \in C_i^*} (d(j, c_i^*) - t_\ell)^+$. Using these facts, we simplify the bound given above:

$$\sum_{j \in \mathcal{C}} \mathbb{E}[\min(\delta_j, f)] \leq 9kf + 14\text{OPT} + 3B \tag{6.2}$$

We note that the above bound is independent of the conditioning, which can hence be removed. We can use (6.2) to establish an upper bound on the expected (connection) cost induced by our solution S, as well as the expected size of S. Recall that $\text{Top}_\ell(d(\mathcal{C}, S)) \leq \ell \cdot 3t_\ell + \sum_{j \in \mathcal{C}} (d(j, S) - 3t_\ell)^+$, and $f = \frac{B}{k}$. We have

$$\begin{aligned}
\mathbb{E}\left[\text{Top}_\ell(d(\mathcal{C}, S))\right] &\leq \mathbb{E}\left[\ell \cdot 3t_\ell + \sum_{j \in \mathcal{C}} (d(j, S) - 3t_\ell)^+\right] \\
&\leq 3B + \sum_{i=1}^{k} \sum_{j \in C_i^*} \mathbb{E}[\min(\delta_j, f)] \\
&\leq 3B + 9kf + 14\text{OPT} + 3B \leq 15B + 14\text{OPT}
\end{aligned}$$

We can also derive the following bound on the expected size of S.

$$\sum_{i=1}^{k} \mathbb{E}[|S \cap C_i^*|] \leq \sum_{i=1}^{k} \sum_{p \in C_i^*} \frac{\mathbb{E}[\min(\delta_p, f)]}{f} \leq \frac{9kf + 14\text{OPT} + 3B}{f} \leq 26k$$

□

Mechanism 9: $O(1)$-distortion, $O((\log(1/\delta)+\log k)\log n)$-query mechanism for ℓ-centrum

Input: Preference profile σ

$B \leftarrow n \cdot \sum_{e\in T} d^*(e)$, where T is the min cost k-Forest returned by Algorithm 4
Obtain a random sequence of agents $x_1, \ldots, x_n$ via shuffling
$\mathcal{S} \leftarrow \varnothing$
for $i = 1, \ldots, \lceil \log_2 n^2 \rceil + 1$ **do**
    $L_i \leftarrow 2^{i-1} \cdot B/n$
    $f \leftarrow \frac{L_i}{k}$
    repeat $\log(1/\delta)$ **times**
        $S \leftarrow \{x_1\}$
        for $j = 2, \ldots, n$ **do**
            Query x_j for the value of $d^*(x_j, top_S(x_j))$
            $\delta_j = \left(d^*(x_j, top_S(x_j)) - 3 \cdot \frac{L_i}{\ell}\right)^+$
            Add x_j to S with probability $\min(1, \delta_j/f)$
        end
        if $|S| \leq 108k$ **then**
            $\mathcal{S} \leftarrow \mathcal{S} \cup \{S\}$
        end
    end
end
Choose $S \in \arg\min_{S'\in\mathcal{S}} \mathrm{Top}_\ell(d^*(\mathcal{C}, S'))$; $\{P_j\}_{j\in S}$ is the partition induced by S
For $j \in S$, $w_j \leftarrow |P_j|$
$\overline{S} \leftarrow$ Algorithm7$(S, \sigma, \{w_j\}_{j\in S}, B, \mathcal{A})$ where $\mathcal{A}$ is an $O(1)$-approximation algorithm for ℓ-centrum
return $\overline{S}$

Theorem 6.2.4. *Mechanism 9 is an $O(1)$-distortion mechanism for ℓ-centrum that requires $O((\log(1/\delta)+\log k)\log n)$ value queries per agent, and has a success probability of at least $1-\delta$.*

Proof. Let $\varepsilon \in (0, 1]$ be a constant. Notice that in lines 7-17, we are running Algorithm 8 $\log(1/\delta)$ times for a given L_i (which serves as our estimate for OPT). Moreover, since we know that $\text{OPT} \leq B \leq n^2 \cdot \text{OPT}$, there exists some $i^* \in \{1, \ldots, \lceil \log_2 n^2 \rceil\}$ such that $\text{OPT} \leq L_{i^*} \leq 2 \cdot \text{OPT}$.

It suffices to show that the with probability at least $1 - \delta$, one of the solutions returned by Meyerson's algorithm when $f = L_{i^*}/k$ opens at most $108k$ centers, and induces a total connection cost of at most $180\text{OPT}(d^*)$. By Theorem 6.2.1 and Markov's inequality, the Top_ℓ cost of the output of Algorithm 8 when $f = L_{i^*}/k$ is at most $4 \cdot 44\text{OPT}$ with a probability of $\frac{3}{4}$; and the number of centers opened is at most $4 \cdot 26k = 104k$, with a probability of $\frac{3}{4}$. Hence, by Union bound, with a probability of at least $\frac{1}{2}$, the solution returned by Meyerson's algorithm is a (176, 104)-bicriteria approximate k-median solution. Since we run Meyerson's algorithm $\log(1/\delta)$ times, with probability at least $1 - \delta$, there exists $S \in \mathcal{S}$ such that $\text{Top}_\ell(d^*(\mathcal{C}, S)) \leq 176 \cdot \text{OPT}$ and $|S| \leq 104k$. As noted before, this S yields a sparsified (weighted) instance.

We would like to apply the blackbox reduction (Algorithm 7) to this sparsified instance. However, as before, it is possible that OPT', the optimal value of the weighted instance is less than OPT, and hence $\text{OPT}' \leq B \leq n \cdot \text{OPT}'$ may not hold. By Lemma 6.0.1, $\text{OPT}' \leq 2(176 + 1)\text{OPT}$; if we take $U = 354\text{OPT}$, $U \leq 354 \leq n \cdot U$, and hence we can apply Theorem 6.1.4. This yields a $354(\rho(1 + 2\varepsilon) + \varepsilon)$-approximate solution to the original instance, where ρ is the approximation factor of the k-median algorithm used in step 21. In particular, if we use Chakrabarty and Swamy's $(5 + \varepsilon)$-approximation algorithm for the ℓ-centrum problem, we obtain a solution of cost at most $354((5 + \varepsilon)(1 + 2\varepsilon) + \varepsilon)\text{OPT}(d^*)$.

Query Complexity: The steps which require value queries are lines 4-18, line 19, and the call to Algorithm 7 in line 21. The total number of queries made in lines 4-18 is $O(\log(1/\delta) \cdot \log(n))$. Since $|\mathcal{S}| = O(\log(1/\delta) \cdot \log n)$, the number of queries per agent to find $S \in \arg\min_{S' \in \mathcal{S}} \text{Top}_\ell(d^*(\mathcal{C}, S'))$ is at most $O(\log(1/\delta) \cdot \log n)$ queries per agent. Finally, since the weighted instance given as input to Algorithm 7 in line 21 consists of $O(k)$ points, $B \in [\text{OPT}, n^2 \cdot \text{OPT}]$, this step takes at most $O(\log n \log k)$ queries per agent (by Theorem 6.1.4). □

6.3 Adaptive sampling for ℓ-centrum k-clustering

The core of Mechanism 6 is Aggarwal, Deshpande, and Kannan's adaptive sampling algorithm for k-median clustering (Algorithm 2). A natural question is whether this adaptive sampling algorithm yields a constant-factor bicriteria approximation for the more general ℓ-centrum. If this were true, then Algorithm 2 would also yield a constant-factor bicriteria approximation for k-center, as it is a special case of ℓ-centrum when $\ell = 1$. However, as illustrated by Theorem 6.3.1, if $\tau = O(1)$, even for $k = 2$ there exist instances for which the Algorithm 2 can be arbitrarily bad for k-center.

Theorem 6.3.1. *For any constant $\tau \geq 1$, $L > 1$, $\epsilon > 0$, there exists an instance $\mathcal{I} = (\mathcal{C}, d, k)$ such that $\Pr[Top_1(d(\mathcal{C}, S)) < L \cdot OPT] < 2\epsilon$, where S is the set of centers τk opened by running Algorithm 2 on $\mathcal{I}$ and OPT is the value of an optimal k-center solution for the instance $\mathcal{I}$.*

Proof. Let the set of agents be $\mathcal{C} = C_1 \cup \{j^*\}$, where $|C_1| > 2\tau + \frac{1}{\epsilon} \cdot 2\tau L$. For all $i, j \in C_1$, $d(i, j) = 1$, and for all $j \in C_1$, $d(j, j^*) = L$. Notice that this defines a valid metric. Fix $k = 2$. An optimal solution for 2-center would be to open one center in C_1, and one center at j^*; this solution has a cost of 1, so OPT $= 1$. For any $S \subseteq \mathcal{C}$, if $\text{Top}_1(d(\mathcal{C}, S)) < L = L \cdot \text{OPT}$, then $d(j^*, S) < L$; but since $d(i, j^*) = L$ for all $i \in \mathcal{C} \setminus \{j^*\}$, this is only possible if $j^* \in S$.

Let S_{i-1} be the set of centers opened by the end of step $i - 1$ of the d-sampling algorithm, and let s_i be the center opened in step i. $\Pr[s_i = j^* | j^* \notin S_{i-1}] = \frac{L}{|C_1| - |S_{i-1}| + L} \leq \frac{L}{|C_1| - 2\tau + L} < \frac{\epsilon}{2\tau}$. By Union bound, $\Pr[j^* \in S | j^* \notin S_1] < |S| \cdot \frac{\epsilon}{2\tau} = \epsilon$. Assuming that the first center is chosen uniformly at random, $\Pr[j^* \notin S_1] = \frac{n-1}{n}$, where $n = |\mathcal{C}|$, so $\Pr[j^* \in S] = \Pr[j^* \in S_1] + \Pr[j^* \in S | j^* \notin S_1] \cdot \Pr[j^* \notin S_1] < \frac{1}{n} + \epsilon < 2\epsilon$. Hence, $\Pr[\text{Top}_1(d(\mathcal{C}, S)) < L \cdot \text{OPT}] \leq \Pr[j^* \in S] < 2\epsilon$. □

Unlike k-center, in k-median, each agent contributes exactly $d(j, S)$ to the objective. Intuitively, Algorithm 2 works for k-median because we are sampling the next center to open with probability proportional to this individual contribution. This, however, is not the case when we try to use Algorithm 2 for k-center, as illustrated by the instance given

in the proof of Theorem 6.3.1. In this instance, if the first center is opened in C_1, any good algorithm should open the other center at j^*, as after the first center is opened in C_1, agents in C_1 no longer contribute to the cost of the current solution (until after a center at j^* is opened). However, as argued previously, Algorithm 2 fails to do this with non-negligable probability. In general, the ℓ-centrum objective is not so easily separable into individual contributions. As only the agents corresponding to the ℓ largest assignment costs contribute to the objective, the contribution of an agent depends on the assignment costs incurred by the other agents. In order to design a good adaptive sampling algorithm for the ℓ-centrum problem, we must first overcome this inherent dependence on the relative ordering of the agents with respect to assignment cost. As a means to this end, we will Chakrabarty and Swamy's proxy function [20].

Notice that the contribution of an agent j to $\ell \cdot \beta t_\ell^* + \sum_{j \in \mathcal{C}} (d(j,S) - \beta t_\ell^*)^+$ can be viewed as $(d(j,S) - \beta t_\ell^*)^+$. Thus, Chakrabarty and Swamy's proxy function eliminates the dependence on the relative ordering of the agents, and allows us to decompose an upper bound on the objective value of a given solution S in terms of "individual contributions" of the agents. So, at each step of our adaptive sampling algorithm, we sample the next center to open with probability proportional to this contribution. For ease of exposition, we present the adaptive sampling algorithm under the assumption that t_ℓ^* is known; later, we will show how to relax this assumption. We will show that this adaptive sampling algorithm gives a constant factor bicriteria approximation for the ℓ-centrum problem. Let β, τ be some parameters that we will choose later.

Algorithm 10: Adaptive sampling algorithm for ℓ-centrum

Input: An ℓ-centrum instance $(\mathcal{C}, d)$
t_ℓ: a guess for t_ℓ^*

$S_0 \leftarrow \varnothing$;
for $i = 1, \ldots, \tau(k + \sqrt{k})$ **do**
  Sample s_i with probability proportional to $(d(s_i, S_{i-1}) - \beta t_\ell)^+$
  Update $S_i \leftarrow S_{i-1} \cup \{s_i\}$.
end
Return $S_{\tau(k+\sqrt{k})}$

Theorem 6.3.2. *Let S be the set of centers opened by Algorithm 10 and suppose that $t_\ell^* \le t_\ell \le \max\{(1+\varepsilon)t_\ell^*, \varepsilon \cdot \frac{OPT}{\ell}\}$. Let ρ be a constant that is strictly larger than 30. Then, there exists suitable parameters such that $|S| = O(k)$ and with constant probability, $Top_\ell(d(\mathcal{C}, S)) \le \rho \cdot OPT$.*

Proof. Let $\alpha, \beta, \gamma, \kappa$ be constants that we will choose later, and define $\tau = \left(\left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right) \cdot \frac{\alpha-1}{2\kappa\alpha}\right)^{-1}$. We wish to show that after opening $\tau(k + \sqrt{k})$ centers, $\mathrm{Top}_\ell(d(\mathcal{C}, S)) \le \rho \cdot \mathrm{OPT}$, with constant probability. We outline the proof approach here. Let $C_1^*, \ldots, C_k^*$, with centers $c_1^*, \ldots, c_k^*$ respectively, be the k clusters induced by the optimal solution, S^*. If we could show that for every cluster C_q^*, $\sum_{j \in C_q^*}(d(j,S) - \beta t_\ell)^+ \le \gamma \sum_{j \in C_q^*}(d(j, c_q^*) - t_\ell)^+$, then we would be able to show that the Top_ℓ cost of S is within a constant factor of OPT. To be precise,

Definition 6.3.3. A cluster C_q^* is *ℓ-good*, if $\sum_{j \in C_q^*}(d(j,S) - \beta t_\ell)^+ \le \gamma \sum_{j \in C_q^*}(d(j, c_q^*) - t_\ell)^+$

Claim 6.3.4. *If every cluster is ℓ-good,*

$$Top_\ell(d(\mathcal{C}, S)) \le \max\{(1+\varepsilon)\beta, (\gamma+\varepsilon)\} \cdot Top_\ell(d(\mathcal{C}, S^*))$$

Proof. If $t_\ell \le (1+\varepsilon)t_\ell^*$,

$$\begin{aligned}
\mathrm{Top}_\ell(d(\mathcal{C}, S)) &\le \ell \cdot \beta t_\ell + \sum_{j \in \mathcal{C}}(d(j,S) - \beta t_\ell)^+ \\
&\le \ell \cdot \beta(1+\varepsilon)t_\ell^* + \gamma \sum_{j \in \mathcal{C}}(d(j, S^*) - t_\ell^*)^+ \\
&\le \max\{(1+\varepsilon)\beta, \gamma\} \cdot \mathrm{Top}_\ell(d(\mathcal{C}, S^*))
\end{aligned}$$

Otherwise, $t_\ell \le \varepsilon \cdot \frac{\mathrm{OPT}}{\ell}$. Then,

$$\begin{aligned}
\mathrm{Top}_\ell(d(\mathcal{C}, S)) &\le \ell \cdot \beta t_\ell + \sum_{j \in \mathcal{C}}(d(j,S) - \beta t_\ell)^+ \\
&\le \ell \cdot \frac{\varepsilon \mathrm{OPT}}{\ell} + \gamma \sum_{j \in \mathcal{C}}(d(j, S^*) - t_\ell^*)^+ \\
&\le (\gamma + \varepsilon) \cdot \mathrm{Top}_\ell(d(\mathcal{C}, S^*))
\end{aligned}$$

□

So, we wish to show that, at termination, all clusters are ℓ-good with constant probability. If a cluster is not ℓ-good, we say that it is *ℓ-bad*. Notice that if a cluster is ℓ-good in step i, it remains ℓ-good in all subsequent steps. We will show that, with constant probability, we turn at least one ℓ-bad cluster into an ℓ-good cluster in each step.

Let s_i be the center opened in step i, and let C^* be the optimal cluster such that $s_i \in C^*$; furthermore, let c^* be the center of C^*. We define $r_\ell(C^*) = \frac{\sum_{j \in C^*}(d(j,c^*)-t_\ell)^+}{|C^*|}$. There are two cases to consider here. If c^* is close to one of the currently open centers, we say that C^* is *ℓ-close*; otherwise, we say that C^* is ℓ-far.

Definition 6.3.5. Let C_q^* be a cluster with center c_q^*. If $d(c_q^*, S_i) \leq \kappa \cdot \max\{t_\ell, r_\ell(C_q^*)\}$, we say that the cluster C^* is ℓ-close at step i; otherwise, C_q^* is ℓ-far.

As done by Aggarwal et. al, we will show that, with constant probability, a point is selected from the core of a bad cluster. However, unlike Aggarwal et. al, we define the core of a cluster differently, depending on if it is an ℓ-close or ℓ-far cluster.

Definition 6.3.6. The ℓ-core of an ℓ-close cluster C_q^*, whose center is c_q^*, is $core_\ell(C_q^*) = \{j \in C_q^* : d(j, c_q^*) \leq t_\ell\}$. The ℓ-core of an ℓ-far cluster C_q^* is $core_\ell(C_q^*) = \{j \in C_q^* : (d(j, c_q^*) - t_\ell)^+ \leq \alpha \cdot r_\ell(C_q^*)\}$

When the context is clear, we will refer to the ℓ-core of C^* as simply the core of C^*. If s_i belongs to the core of C^*, C^* becomes ℓ-good, as $\sum_{j \in C^*}(d(j, s_i) - \beta t_\ell)^+ \leq \sum_{j \in C^*}(d(j, c^*) - t_\ell)^+ + |C^*|(d(s_i, c^*) - t_\ell)^+ \leq \sum_{j \in C^*}(d(j, c^*) - t_\ell)^+ + \alpha |C^*| r_\ell(C^*) \leq (1 + \alpha) \sum_{j \in C^*}(d(j, c^*) - t_\ell)^+$, where the second inequality is because $s_i \in core_\ell(C^*)$. Hence, we want to show that, with constant probability, s_i belongs to the core of an ℓ-bad cluster C^*.

Lemma 6.3.7. *Let S_{i-1} be the set of currently opened centers, and let s_i be sampled with probability proportional to $(d(s_i, S_{i-1}) - \beta t_\ell)^+$. Then, $Top_\ell(d(\mathcal{C}, S_{i-1})) \leq \rho \cdot Top_\ell(d(\mathcal{C}, S^*))$, or s_i is in the ℓ-core of an ℓ-bad cluster, with constant probability.*

Proof of Lemma 6.3.7. If $\text{Top}_\ell(d(\mathcal{C}, S_{i-1})) \leq \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$, we are done – so assume that $\text{Top}_\ell(d(\mathcal{C}, S_{i-1})) > \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$. We first show that with constant probability, s_i is in an ℓ-bad cluster.

Claim 6.3.8. $\Pr[s_i \in \ell\text{-good cluster}] \leq \frac{\max\{\beta,\gamma\}}{\rho}$

Proof of Claim 6.3.8. By definition of ℓ-good clusters,

$$\sum_{C^* \ \ell\text{-good}} \sum_{j \in C^*} (d(j, S_{i-1}) - \beta t_\ell)^+ \leq \sum_{C^* \ \ell\text{-good}} \gamma \cdot \sum_{j \in C^*} (d(j, c^*) - t_\ell)^+$$

so,

$$\begin{aligned}
\Pr[s_i \in \ell\text{-good cluster}] &= \frac{\sum_{C^* \text{ is } \ell\text{-good}} \sum_{j \in C^*} (d(j, S_{i-1}) - \beta t_\ell)^+}{\sum_{j \in \mathcal{C}} (d(j, S_{i-1}) - \beta t_\ell)^+} \\
&\leq \frac{\beta t_\ell \cdot \ell + \sum_{C^* \text{ is } \ell\text{-good}} \gamma \cdot \sum_{j \in C^*} (d(j, c^*) - t_\ell)^+}{\beta t_\ell \cdot \ell + \sum_{j \in \mathcal{C}} (d(j, S_{i-1}) - \beta t_\ell)^+} \\
&\leq \frac{\max\{\beta, \gamma\} \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))}{\rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))} = \frac{\max\{\beta, \gamma\}}{\rho}
\end{aligned}$$

where the third inequality is because $\beta t_\ell \cdot \ell + \sum_{j \in \mathcal{C}} (d(j, S_{i-1}) - \beta t_\ell)^+ \geq \text{Top}_\ell(d(\mathcal{C}, S_{i-1})) > \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$ by Claim 2.2.1. □

So, with constant probability, C^* is an ℓ-bad cluster. If C^* is ℓ-close, we can show that $s_i \in \mathit{core}_\ell(C^*)$ with constant probability.

Claim 6.3.9. $\Pr[C^* \text{ is } \ell\text{-close}, s_i \notin \mathit{core}_\ell(C^*)] \leq \frac{\kappa+\beta}{\rho}$

Proof of Claim 6.3.9. If C^* is ℓ-close, $\mathit{core}_\ell(C^*) = \{j \in C^* : d(j, c^*) \leq t_\ell\}$. As t_ℓ is at least the value of t^*_ℓ, the ℓth largest assignment cost induced by the optimal solution,

$$|\{j \in \mathcal{C} : d(j, S^*) > t_\ell\}| \leq |\{j \in \mathcal{C} : d(j, S^*) > t^*_\ell\}| \leq \ell$$

so the probability that $s_i \notin \mathit{core}_\ell(C^*)$ is small.

$$\begin{aligned}
&\Pr[C^* \ \ell\text{-close}, d(s_i, c^*) > t_\ell] \\
&= \frac{\sum_{C^*_q \ \ell\text{-close}} \sum_{j \notin \mathit{core}_\ell(C^*_q)} (d(j, S_{i-1}) - \beta t_\ell)^+}{\sum_{j \in \mathcal{C}} (d(j, S_{i-1}) - \beta t_\ell)^+} \\
&\leq \frac{\ell \cdot \beta t_\ell + \sum_{C^*_q \ \ell\text{-close}} \sum_{j \notin \mathit{core}_\ell(C^*_q)} (d(j, c^*_q) + d(c^*_q, S_{i-1}) - \beta t_\ell)^+}{\text{Top}_\ell(d(\mathcal{C}, S_{i-1}))} \\
&\leq \frac{\ell \cdot \beta t_\ell + \sum_{C^*_q \ \ell\text{-close}} |C^*_q \setminus \mathit{core}_\ell(C^*_q)| \cdot (d(c^*, S_{i-1}) - t_\ell)^+}{\text{Top}_\ell(d(\mathcal{C}, S_{i-1}))} \\
&+ \frac{\sum_{C^*_q \ \ell\text{-close}} \sum_{j \notin \mathit{core}_\ell(C^*_q)} (d(j, c^*_q) - t_\ell)^+}{\text{Top}_\ell(d(\mathcal{C}, S_{i-1}))}
\end{aligned}$$

Notice that $\sum_{C_q^* \ \ell\text{-close}} |C_q^* \setminus core_\ell(C_q^*)| \leq |\{j \in \mathcal{C} : d(j, S^*) > t_\ell\}| \leq \ell$. Furthermore, for every ℓ-close cluster C_q^*, $d(c_q^*, S_{i-1}) \leq \kappa \max\{t_\ell, r_\ell(C_q^*)\}$, so

$$\begin{aligned}\sum_{C_q^* \ \ell\text{-close}} |C_q^* \setminus core_\ell(C_q^*)| \cdot (d(c_q^*, S_{i-1}) - t_\ell)^+ &\leq \ell \cdot \kappa t_\ell + \kappa \sum_{C_q^* \ \ell\text{-close}} \sum_{j \in C_q^*} (d(j, c_q^*) - t_\ell)^+ \\ &\leq \kappa \cdot \mathrm{Top}_\ell(d(\mathcal{C}, S^*))\end{aligned}$$

Thus,

$$\Pr[C^* \text{ is } \ell\text{-close}, s_i \notin core_\ell(C^*)] \leq \frac{(\kappa + \beta) \cdot \mathrm{Top}_\ell(d(\mathcal{C}, S^*))}{\rho \cdot \mathrm{Top}_\ell(d(\mathcal{C}, S^*))} = \frac{\kappa + \beta}{\rho}$$

□

The other case is that C^* is ℓ-far.

Claim 6.3.10. $\Pr[s_i \in core_\ell(C^*) | C^* \ \ell\text{-far}, C^* \ \ell\text{-bad}] \geq \frac{\alpha - 1}{\alpha\kappa}$

Proof of Claim 6.3.10. If C^* is ℓ-far, $core_\ell(C^*) = \{j \in C^* : (d(j, c^*) - t_\ell)^+ \leq \alpha \cdot r_\ell(C^*)\}$. As

$$|C^*| \cdot r_\ell(C^*) \geq \sum_{j \notin core_\ell(C^*)} (d(j, c^*) - t_\ell)^+ \geq |C^* \setminus core_\ell(C^*)| \cdot \alpha r_\ell(C^*)$$

we have that $|core_\ell(C^*)| \geq \frac{\alpha - 1}{\alpha} \cdot |C^*|$. Combining this fact with the triangle inequality yields

$$\begin{aligned}\Pr[s_i \in core_\ell(C^*) | C^* \ \ell\text{-far}, C^* \ \ell\text{-bad}] &= \frac{\sum_{j \in core_\ell(C^*)} (d(j, S_{i-1}) - \beta t_\ell)^+}{\sum_{j \in C^*} (d(j, S_{i-1}) - \beta t_\ell)^+} \\ &\geq \frac{\sum_{j \in core_\ell(C^*)} (d(c^*, S_{i-1}) - d(j, c^*) - \beta t_\ell)^+}{\sum_{j \in C^*} (d(j, c^*) + d(c^*, S_{i-1}) - \beta t_\ell)^+} \\ &\geq \frac{|core_\ell(C^*)| \cdot (d(c^*, S_{i-1}) - \alpha r_\ell - (\beta + 1) t_\ell)}{|C^*| \cdot (r_\ell + (d(c^*, S_{i-1}) - t_\ell))} \\ &\geq \frac{\alpha - 1}{\alpha} \cdot \frac{d(c^*, S_{i-1}) - \alpha r_\ell - (\beta + 1) t_\ell}{r_\ell + d(c^*, S_{i-1}) - t_\ell}\end{aligned}$$

The third inequality is because $d(j, c^*) - t_\ell \leq r_\ell$ for all $j \in core_\ell(C^*)$. Note that

$$\frac{d(c^*, S_{i-1}) - \alpha r_\ell(C^*) - (\beta + 1) t_\ell}{r_\ell(C^*) + d(c^*, S_{i-1}) - t_\ell}$$

is an increasing function of $d(c^*, S_{i-1})$. Since C^* is an ℓ-far cluster, $d(c^*, S_{i-1}) > \kappa \cdot \max\{t_\ell, r_\ell(C^*)\}$, so

$$\frac{d(c^*, S_{i-1}) - \alpha r_\ell(C^*) - (\beta+1)t_\ell}{r_\ell(C^*) + d(c^*, S_{i-1}) - t_\ell} \geq \frac{\kappa \cdot \max\{t_\ell, r_\ell(C^*)\} - \alpha r_\ell(C^*) - (\beta+1)t^*_\ell}{r_\ell(C^*) + \kappa \cdot \max\{t_\ell, r_\ell(C^*)\} - t_\ell} \geq \frac{1}{\kappa}$$

□

Finally, we can establish a lower bound on $\Pr[C^* \ \ell\text{-bad}, s_i \in core_\ell(C^*)]$ using Claims 6.3.8 - 6.3.10.

$$\begin{aligned}
&\Pr[C^* \ \ell\text{-bad}, s_i \in core_\ell(C^*)] \\
&= \Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-close}] - \Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-close}, s_i \notin core_\ell(C^*)] \\
&\quad + \Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-far}, s_i \in core_\ell(C^*)]
\end{aligned}$$

Note that $\Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-close}] + \Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-far}] = \Pr[C^* \ \ell\text{-bad}] \geq 1 - \frac{\max\{\beta,\gamma\}}{\rho}$. Suppose $\Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-close}] \geq \frac{1}{2} \cdot \left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right)$. Then,

$$\frac{1}{4} \cdot \left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right) \geq \frac{(\kappa+\beta)}{\rho} \geq \Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-close}, s_i \notin core_\ell(C^*)],$$

and hence we have

$$\begin{aligned}
&\Pr[C^* \ \ell\text{-bad}, s_i \in core_\ell(C^*)] \\
&\geq \Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-close}] - \Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-close}, s_i \notin core_\ell(C^*)] \\
&\geq \frac{1}{4} \cdot \left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right).
\end{aligned}$$

Otherwise, $\Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-far}] \geq \frac{1}{2} \cdot \left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right)$, and hence

$$\begin{aligned}
&\Pr[C^* \ \ell\text{-bad}, s_i \in core_\ell(C^*)] \\
&\geq \Pr[s_i \in core_\ell(C^*) | C^* \ \ell\text{-bad}, C^* \ \ell\text{-far}] \cdot \Pr[C^* \ \ell\text{-bad}, C^* \ \ell\text{-far}] \\
&\geq \left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right) \cdot \frac{\alpha - 1}{2\kappa\alpha}
\end{aligned}$$

Thus,

$$\Pr[C^* \text{ } \ell\text{-bad}, s_i \in core_\ell(C^*)] \geq \left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right) \cdot \frac{\alpha - 1}{2\kappa\alpha}$$

Moreover, this argument holds for any choice of positive constants $\alpha, \beta, \gamma, \kappa$ that satisfies $\alpha > 1$, $\gamma \geq \alpha + 1$, $\beta \geq 2$, $\frac{1}{4}\left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right) \geq \frac{\kappa+\beta}{\rho}$, $\kappa \geq \alpha + \beta + 2$, and $\rho \geq \max\{(1+\varepsilon)\beta, (\gamma+\varepsilon)\}$. □

Thus, we have shown that $\text{Top}_\ell(d(\mathcal{C}, S_{i-1})) \leq \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$, or s_i is in the ℓ-core of an ℓ-bad cluster, with constant probability. Let Bad_i^ℓ is the set of ℓ-bad clusters in step i; since opening a center in the ℓ-core of a cluster causes the cluster to become ℓ-good, if $\text{Top}_\ell(d(\mathcal{C}, S_{i-1})) > \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$, then by Lemma 6.3.7, $|\text{Bad}_i^\ell| > |\text{Bad}_{i+1}^\ell|$, with constant probability. Furthermore, using Aggarwal et. al's argument [3], we can show that after $\tau(k + \sqrt{k})$ centers are opened, $|\text{Bad}_{\tau(k+\sqrt{k})}^\ell| = 0$ with constant probability.

Lemma 6.3.11 (Aggarwal et. al [3] (Paraphrased)). *Let $B_0 \supseteq B_1 \supseteq \cdots \supseteq B_{(k+\sqrt{k})/p}$ be a chain of subsets such that $B_0 = \{C_1^*, \ldots, C_k^*\}$, and $\Pr[|B_i| > |B_{i+1}|] \geq p$. Then, $\Pr[|B_{(k+\sqrt{k})/p}| = 0] \geq 1 - e^{-p/4}$*

Proof of Lemma 6.3.11 (Adapted from [3]). We wish to prove that, with constant probability, $B_{(k+\sqrt{k})/p} = \varnothing$. For each step i, define a binary random variable X_i, where $X_i = 1$ if $|B_i| = |B_{i+1}|$ and 0 otherwise. Note that $E[X_i] \leq 1-p$. We define $J_i := \sum_{j=1}^{i}(X_j - (1-p))$. Since $J_{i+1} - J_i \leq 1$ and $E[J_i | J_0, \ldots, J_{i-1}] \leq J_{i-1}$, $J_0, \ldots, J_{(k+\sqrt{k})/p}$ are super-martingale. So, we can apply Asuma-Hoeffding's inequality to obtain that $\Pr[J_{(k+\sqrt{k})/p} \geq J_0 + \delta] \leq \exp\{-\delta^2/2((k+\sqrt{k})/p)\}$. Thus, $\Pr\left[\sum_{j=1}^{(k+\sqrt{k})/p}(1 - X_j) \geq k\right] \geq 1 - \exp\left\{-\frac{pk^2}{2(k+\sqrt{k})}\right\} \geq 1 - e^{-p/4}$.

Notice that if $\sum_{j=1}^{(k+\sqrt{k})/p}(1 - X_j) \geq k$, there are at least k steps where $X_j = 0$; since $|B_0| \leq k$ and each time $X_j = 0$, $|B_j| - 1 \geq |B_{j+1}|$, this implies that $B_{(k+\sqrt{k})/p} = \varnothing$. □

Thus, either $\text{Top}_\ell(d(\mathcal{C}, S_{\tau(k-\sqrt{k})})) \leq \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$ or, by Lemma 6.3.11, there are no ℓ-bad clusters with constant probability. □

In particular, if $t_\ell^* \leq t_\ell \leq \max\{33t_\ell^*, \frac{32\text{OPT}}{\ell}\}$ and we set $\alpha = 1.92, \beta = 2, \gamma = 2.92, \kappa = 5.92$, and $\tau = 28$, we get a $(35, 28(1 + 1/\sqrt{k}))$-bicriteria approximation.

Corollary 6.3.12. *If $t_\ell^* \leq t_\ell \leq \max\{33t_\ell^*, \frac{32OPT}{\ell}\}$, $\beta = 2$, and $\tau = 28$, Algorithm 10 yields a solution that opens $28(k + \sqrt{k})$ centers and has cost at most $35 \cdot OPT$, with constant probability.*

It remains to show how to compute t_ℓ such that $\tilde{t}_\ell \in \mathcal{T}$ such that $t_\ell^* \leq \tilde{t}_\ell \leq \max\{(1+\varepsilon)t_\ell^*, \varepsilon \cdot \frac{\mathrm{OPT}}{\ell}\}$. Let t_1^* be the value of the largest assignment cost induced by the optimal k-center solution. Then, $t_1^* \leq \mathrm{OPT} \leq \ell \cdot t_1^*$, so $t_\ell^* \leq \mathrm{OPT} \leq \ell \cdot t_1^*$ and $\varepsilon \cdot \frac{t_1^*}{\ell} \leq \varepsilon \cdot \frac{\mathrm{OPT}}{\ell} \leq \varepsilon \cdot t_1^*$. Hence, there exists $\tilde{t}_\ell \in \{\ell t_1^* \cdot (1+\varepsilon)^{-r} : r = 0, \ldots, \log_{1+\varepsilon}(\frac{\ell^2}{\varepsilon})\}$ such that $t_\ell^* \leq \tilde{t}_\ell \leq \max\{(1+\varepsilon)t_\ell^*, \varepsilon \cdot \frac{\mathrm{OPT}}{\ell}\}$. The benefit of defining a set of guesses in terms of t_1^* is that we can easily compute a 2-approximate k-center solution by repeatedly opening a center at the agent that is the farthest away from the currently open centers.

Mechanism 11: $O(1)$-distortion, $O(k \log(1/\delta) \log \ell)$-query mechanism for ℓ-centrum

$S_1 \leftarrow \varnothing$
repeat k **times**
  for $j \in \mathcal{C}$ **do**
    Query j for the value of $d^*(j, top_{S_1}(j))$
  end
  Choose $s \in \arg\max_{j \in \mathcal{C}} d^*(j, top_{S_1}(j))$ and update $S_1 \leftarrow S_1 \cup \{s\}$
end
$t_1 \leftarrow \max_{j \in \mathcal{C}} d^*(j, S_1)$
$\mathcal{S} \leftarrow \varnothing$
for $t_\ell \in \mathcal{T} = \{\ell t_1 \cdot (1+\varepsilon)^{-r} : r = 0, \ldots, \log_{1+\varepsilon}(\frac{2\ell^2}{\varepsilon})\}$ **do**
  repeat $\log(1/\delta)$ **times**
    $S \leftarrow \varnothing$
    for $i = 1, \ldots, 28(k + \sqrt{k})$ **do**
      for $j \in \mathcal{C}$ **do**
        Query j for the value of $d^*(j, top_S(j))$
      end
      Sample s_i with probability proportional to $(d^*(s_i, S) - 2t_\ell)^+$
      Update $S \leftarrow S \cup \{s_i\}$
    end
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{S\}$
  end
end
Choose $S \in \arg\min_{S' \in \mathcal{S}} \sum_{j \in \mathcal{C}} d^*(j, S')$; $\{P_j\}_{j \in S}$ is the partition induced by S
for $j \in S$ **do**
  Query $d^*(i, j)$ for all $i \in S \setminus \{j\}$
end
$\bar{S}$: output of $(5+\varepsilon)$-approximation algorithm for ℓ-centrum on the (cardinal) weighted instance (S, w, d) where $w_j = |P_j|$ for all $j \in S$
return $\overline{S}$

Theorem 6.3.13. *Mechanism 11 is an $O(1)$-distortion mechanism for ℓ-centrum that requires $O(k \log(1/\delta) \log \ell)$ value queries per agent, and has a success probability of at least $1 - \delta$.*

Proof. In lines 2-6, we run the 2-approximation algorithm for k-center to compute t_1. Since $\frac{t_1}{2} \leq \text{OPT} \leq \ell \cdot t_1$, there exists $\tilde{t}_\ell \in \mathcal{T}$ such that $t^*_\ell \leq \tilde{t}_\ell \leq \max\{(1+\varepsilon)t^*_\ell, \varepsilon \cdot \frac{\text{OPT}}{\ell}\}$. By Theorem 6.3.2, since we run Algorithm 10 $\log(1/\delta)$ times with $\tilde{t}_\ell$, we will obtain a $(35, 28(1 + \frac{1}{\sqrt{k}}))$-bicriteria approximate ℓ-centrum solution, S, with probability at least $1-\delta$. Given this solution, we query all pairwise distances for $i, j \in S$ and use Chakrabarty and Swamy's algorithm [20] to obtain $\overline{S}$, a $(5+\varepsilon)$-approximate solution to the weighted instance induced by S. By Lemma 6.0.1, $\text{Top}_\ell(d(\mathcal{C}, \overline{S})) \leq (35 + (5+\varepsilon)2(35+1))\text{OPT}(d^*)$.

Query Complexity: The total number of queries per agent in lines 2 - 21 is $O(k + \log(\ell) \cdot \log(1/\delta) \cdot k)$. The total number of value queries made in lines 24-26 is $O(k)$ queries per agent in S (since $|S| = O(k)$). Thus, the total number of queries per agent is $O(\log(1/\delta)k \log \ell)$. □

Chapter 7

Adaptive sampling for minimum-norm k-clustering

We now describe how to extend our approach for ℓ-centrum to obtain a $(\rho, O(1))$-bicriteria approximation for minimum-norm k-clustering. Recall that in this problem, we seek to open a set of centers, S, that minimizes $f(d(\mathcal{C}, S))$, where f is a monotone symmetric norm.

Theorem 7.0.1 (Theorem 2.4 in [27]). *If $x, y \in \mathbb{R}^n_+$ are such that $Top_\ell(x) \leq \alpha \cdot Top_\ell(y) + \beta$ for all $\ell \in [n]$, where $\alpha, \beta \geq 0$, then $h(x) \leq \alpha \cdot h(y) + \beta \cdot h(1, 0, \ldots, 0)$ for any monotone, symmetric norm $h : \mathbb{R}^n \to \mathbb{R}^n_+$.*

Theorem 7.0.1 suggests the following strategy for controlling the f-cost of a solution. For a given $\ell \in [n]$, let $OPT_{\ell\text{-centrum}}$ denote the Top_ℓ-cost of an *optimal* ℓ-centrum solution. If we can construct a solution S such that $\text{Top}_\ell(d(\mathcal{C}, S)) \leq \alpha \cdot OPT_{\ell\text{-centrum}}$ for all $\ell \in [n]$, then $f(d(\mathcal{C}, S)) \leq \alpha \cdot f(d(\mathcal{C}, S'))$, for any $S' \subseteq \mathcal{C} : |S'| \leq k$. In fact, such a set S would be an α-approximate solution with respect to *any* monotone symmetric norm. Kumar and Kleinberg [31] proved that, if $\text{Top}_\ell(d(\mathcal{C}, S)) \leq \alpha \cdot OPT_{\ell\text{-centrum}}$ for all $\ell \in [n]$, $|S| = \Omega(k \log n)$ – so we cannot hope for this property to hold if $O(k)$ centers are opened.

We will still strive to simultaneously control the Top_ℓ cost of our solution for all $\ell \in [n]$ – except, instead of comparing the Top_ℓ-cost of our solution against $OPT_{\ell\text{-centrum}}$, we will

compare it against $\text{Top}_\ell(d(\mathcal{C}, S^*))$, where S^* is the set of centers opened by an optimal solution with respect to the min-norm objective. By the following corollary of Theorem 7.0.1, this weaker property is in fact sufficient to enforce that the value of $f(d(\mathcal{C}, S))$ will not be too large.

Corollary 7.0.2. *Let f be a monotone symmetric norm, and let S^* be the set of centers opened by an optimal k-clustering solution that minimizes f. Let $S \subseteq \mathcal{C}$. If $Top_\ell(d(\mathcal{C}, S)) \leq \rho \cdot Top_\ell(d(\mathcal{C}, S^*))$ for all $\ell \in [n]$, $f(d(\mathcal{C}, S)) \leq \rho \cdot f(d(\mathcal{C}, S^*))$.*

Instead of considering all $\ell \in [n]$, we can restrict our attention to $POS_{n,\delta} := \{\min\{\lceil(1+\delta)^s\rceil, n\} : s \geq 0\}$ (when the context is clear, we will simply write POS). This only causes a loss of a $(1+\delta)$-factor.

Theorem 7.0.3 (Claim 2.6 in [27])**.** *If $Top_\ell(d(\mathcal{C}, S)) \leq \rho \cdot Top_\ell(d(\mathcal{C}, S^*))$ for all $\ell \in POS_{n,\delta}$, $f(d(\mathcal{C}, S)) \leq \rho(1+\delta) \cdot f(d(\mathcal{C}, S^*))$.*

This observation motivates the following naive algorithm: Let S_ℓ be the solution obtained by running Algorithm 10, for every $\ell \in POS$. Then, if $S = \bigcup_{i \in POS} S_i$, $\text{Top}_\ell\left(d\left(\mathcal{C}, S\right)\right) \leq \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$ for all $\ell \in POS$, so $f(d(\mathcal{C}, S)) \leq \rho(1+\delta) \cdot f(d(\mathcal{C}, S^*))$. However, $|S| = \Theta(|POS| \cdot k) = \Theta(k \log n)$, so this naive approach gives a $(O(1), O(\log n))$-bicriteria approximation. To reduce the number of centers opened by this approach, we note the following. First, for any $\ell \in POS$, we can terminate the adaptive sampling algorithm the first time that $\text{Top}_\ell\left(d\left(\mathcal{C}, S_i\right)\right) \leq \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$. Secondly, sometimes opening one center can improve the Top_ℓ-cost of the solution for multiple ℓ simultaneously.

At each step i, we will maintain $\mathcal{L}_i = \{\ell \in POS : \text{Top}_\ell\left(d\left(\mathcal{C}, S_i\right)\right) > \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))\}$. Note that the Top_ℓ-cost of our current solution can only improve in each step, so $\mathcal{L}_i \supseteq \mathcal{L}_{i+1}$. Furthermore, if $\mathcal{L}_i = \varnothing$ in some step, $\text{Top}_\ell\left(d\left(\mathcal{C}, S_i\right)\right) \leq \rho \cdot \text{Top}_\ell(d(\mathcal{C}, S^*))$ for all $\ell \in POS$, so $f(d(\mathcal{C}, S_i)) \leq \rho(1+\delta) \cdot \text{OPT}$, and hence we would be done. Our ultimate goal is to show that, with some constant probability, $\mathcal{L}_i = \varnothing$ after $O(k)$ centers are opened. For any $\ell \in [n]$, let t^*_ℓ be the ℓth largest assignment cost induced by the optimal solution (to the minimum-norm k-clustering problem). We will work with a non-increasing threshold vector $\vec{t}$ that satisfies $t^*_\ell \leq t_\ell \leq \max\{(1+\varepsilon)t^*_\ell, \varepsilon \cdot \text{OPT}_{\ell-centrum}/\ell\}$ for all $\ell \in [n]$.

Recall that a cluster C_q^* is ℓ-good if $\sum_{j\in C_q^*}(d(j,S)-\beta t_\ell)^+ \le \gamma\sum_{j\in C_q^*}(d(j,c_q^*)-t_\ell)^+$ (where β,γ are constants that we fix later). We define Bad_i to be the set of clusters that are ℓ-bad for *some* $\ell\in\mathcal{L}_i$. Notice that it is possible for some cluster $C^*\notin\mathrm{Bad}_i$ to be ℓ-bad for some $\ell\notin\mathcal{L}_i$. In this case, it is neither likely, nor necessary, for us to choose a point in the ℓ-core of C^*, as the Top_ℓ-cost of the current solution is already good, for this particular ℓ. We will show that in any step i, if $\mathcal{L}_i\neq\varnothing$, $|\mathrm{Bad}_i|>|\mathrm{Bad}_{i+1}|$ with some constant probability. Once $|\mathrm{Bad}_i|=0$, it is necessarily true that $\mathcal{L}_i=\varnothing$. In the ℓ-centrum setting, we argued that $|\mathrm{Bad}_i^\ell|>|\mathrm{Bad}_{i+1}^\ell|$ by showing that at each step, we chose a point from the ℓ-core of an ℓ-bad cluster. Now, instead of considering a single ℓ, we wish to show that the center opened in step i is in the ℓ-core of some $C^*\in\mathrm{Bad}_i$ *for all* $\ell\in\mathcal{L}_i$, as then $C^*\notin\mathrm{Bad}_{i+1}$

Lemma 7.0.4. *Let $\ell_i^*=\max_{\ell\in\mathcal{L}_i}\ell$, and let s_i be the center opened in step i. If $s_i\in core_{\ell_i^*}(C^*)$ then C^* becomes ℓ-good for all $\ell\in\mathcal{L}_i$.*

Proof. Consider any $\ell\in\mathcal{L}_i$. If $d(s_i,c^*)\le t_\ell$, then clearly, C^* is ℓ-good. Otherwise, since $s_i\in core_{\ell_i^*}(C^*)$ and $d(s_i,c^*)>t_\ell\ge t_{\ell_i^*}$, C^* is ℓ_i^*-far and ℓ-far, so

$$\begin{aligned}(d(s_i,c^*)-t_\ell)^+ &= (d(s_i,c^*)-t_{\ell_i^*})^+ - (t_\ell-t_{\ell_i^*})\\ &\le \frac{\sum_{j\in C^*}(d(j,c^*)-t_{\ell_i^*})^+}{|C^*|}-(t_\ell-t_{\ell_i^*})\\ &\le \sum_{j\in C^*}\frac{(d(j,c^*)-t_\ell)^+}{|C^*|}\end{aligned}$$

where the second inequality is because $s_i\in core_{\ell_i^*}(C^*)$ and the last inequality is because $(d(j,c^*)-t_\ell)^+\ge(d(j,c^*)-t_{\ell_i^*})^+-(t_\ell-t_{\ell_i^*})$, for any $j\in C^*$. □

Since $\ell_i^*\in\mathcal{L}_{i-1}$, $\mathrm{Top}_{\ell_i^*}(d(\mathcal{C},S_{i-1}))>\rho\cdot\mathrm{Top}_{\ell_i^*}(d(\mathcal{C},S^*))$ – so if s_i is sampled with probability proportional to $(d(s_i,S_{i-1})-\beta t_{\ell_i^*})^+$, by Lemma 6.3.7, s_i lies in the ℓ_i^*-core of

an ℓ_i^*-bad cluster, with constant probability. This motivates the following algorithm.

Algorithm 12: Adaptive sampling algorithm for min-norm k-clustering (when t^* is known)

$S_0 \leftarrow \varnothing$
$\mathcal{L}_1 \leftarrow POS$
for $i = 1, \ldots, \tau(k + \sqrt{k})$ **do**
  $\ell_i^* \leftarrow \max_{\ell \in \mathcal{L}_i} \ell$
  Sample s_i with probability proportional to $(d(s_i, S_{i-1}) - \beta t_{\ell_i^*})^+$
  Update $S_i \leftarrow S_{i-1} \cup \{s_i\}$
  Update $\mathcal{L}_{i+1} \leftarrow \{\ell \in POS : \mathrm{Top}_\ell(d(\mathcal{C}, S_i)) > \rho \cdot \mathrm{Top}_\ell(d(\mathcal{C}, S^*))\}$
end
return $S_{\tau(k+\sqrt{k})}$

Lemma 7.0.5. *If $\mathcal{L}_i \neq \varnothing$, $|Bad_i| > |Bad_{i+1}|$ with constant probability.*

Proof. Note that since $\mathcal{L}_i \neq \varnothing$, ℓ_i^* is well-defined. Let s_i be the new center that is opened in step i. Then, by Lemma 6.3.7, s_i lies in the ℓ_i^*-core of an ℓ_i^*-bad cluster, C^*, with constant probability. Note that $C^* \in \mathrm{Bad}_i$, since C^* is an ℓ_i^*-bad cluster, and $\ell_i^* \in \mathcal{L}_i$. Furthermore, by Lemma 7.0.4, if $s_i \in core_{\ell_i^*}(C^*)$, C^* becomes ℓ-good for all $\ell \in \mathcal{L}_i \supseteq \mathcal{L}_{i+1}$; so $C^* \notin \mathrm{Bad}_{i+1}$. Thus, $|\mathrm{Bad}_i| > |\mathrm{Bad}_{i+1}|$ with constant probability. □

Corollary 7.0.6. *After $\tau(k + \sqrt{k})$ steps, $\mathcal{L}_{\tau(k+\sqrt{k})} = \varnothing$ with constant probability.*

If $\mathrm{Bad}_i = \varnothing$, $\mathcal{L}_i = \varnothing$, so this corollary follows immediately from Lemma 7.0.5 and Lemma 6.3.11.

It remains to show how to compute a threshold vector that is a good guess of $(t_1^*, \ldots, t_n^*)$. Chakrabarty and Swamy proved that one can obtain such a polynomial-sized set of candidate threshold vectors that contain a good guess of $(t_1^*, \ldots, t_n^*)$ [20].

Lemma 7.0.7 (Lemma 6.9 in [20])**.** *Suppose that we can obtain in polynomial time a (polynomial-size) set $A \subseteq \mathbb{R}$ containing a value ρ satisfying $t_1^* \leq \rho \leq (1 + \varepsilon)t_1^*$. Then, in time $O(|A| \cdot |POS| \cdot \max\{(\frac{n}{\varepsilon})^{O(1/\varepsilon)}, n^{1/\delta}\}) = O(|A| \cdot \max\{(\frac{n}{\varepsilon})^{O(1/\varepsilon)}, n^{1/\delta}\})$, we can obtain*

a set $\mathcal{T} \subseteq \mathbb{R}_+^{POS}$ that contains a valid threshold vector $\tilde{t}$ such that $\tilde{t}$ is in non-increasing order, and $t_\ell^* \leq \tilde{t}_\ell \leq (1+\varepsilon)t_\ell^*$ for all $\ell \in POS$.

In fact, we can show that in our setting, there exists a set A of size $O(\frac{1}{\varepsilon}\log n)$ satisfying the conditions of Lemma 7.0.7. As aforementioned, a 2-approximate solution for k-center can be computed by (deterministically) opening a center at the client farthest away from the currently open centers. Let S_1^* be the set of k centers opened by this algorithm. We claim that $\mathrm{Top}_1(d(\mathcal{C}, S^*)) = t_1^* \leq n \cdot \mathrm{Top}_1(d(\mathcal{C}, S_1^*))$. Suppose not. Then, $\mathrm{Top}_\ell(d(\mathcal{C}, S^*)) \geq t_1^* > n \cdot \mathrm{Top}_1(d(\mathcal{C}, S_1^*)) \geq \mathrm{Top}_\ell(d(\mathcal{C}, S_1^*))$ for all $\ell \in [n]$, and so $f(d(\mathcal{C}, S_1^*)) < f(d(\mathcal{C}, S^*))$, a contradiction. So, $\frac{1}{2}\mathrm{Top}_1(d(\mathcal{C}, S_1^*)) \leq t_1^* \leq n \cdot \mathrm{Top}_1(d(\mathcal{C}, S_1^*))$, and hence the set $A = \{\frac{1}{2}\mathrm{top}_1(d(\mathcal{C}, S_1^*)) \cdot (1+\varepsilon)^i : i = 0, 1, \ldots, \lceil \log_{1+\epsilon} 2n \rceil\}$ contains ν such that $t_1^* \leq \nu \leq (1+\varepsilon)t_1^*$.

Corollary 7.0.8. *We can obtain a set $\mathcal{T} \subseteq \mathbb{R}_+^{POS}$ of size $O(\frac{1}{\varepsilon} \cdot \log(n) \cdot \max\{(\frac{n}{\varepsilon})^{O(1/\varepsilon)}, n^{1/\delta}\})$ that contains a valid threshold vector $\tilde{t}$ such that $\tilde{t}$ is in non-increasing order, and $t_\ell^* \leq \tilde{t}_\ell \leq (1+\varepsilon)t_\ell^*$ for all $\ell \in POS$.*

Given a good guess of t^*, we must also be able to compute a good estimate of $\mathrm{Top}_\ell(d(\mathcal{C}, S^*))$ for all $\ell \in POS$. Ibrahimpur and Swamy [27] showed that, given $\tilde{t}$ such that $t_\ell^* \leq \tilde{t}_\ell \leq (1+\varepsilon)t_\ell^*$ for all $\ell \in POS_{n,\delta}$, one can compute a $(1+\delta)(1+\varepsilon)$-approximate guess of $\mathrm{Top}_\ell(d(\mathcal{C}, S^*))$ for any $\ell \in [n]$.

Lemma 7.0.9 (Lemma 2.8 (a) and (b) in [27])**.** *Let $u \in \mathbb{R}_{\geq 0}^m$, and $v \in \mathbb{R}_{\geq 0}^{POS}$ be a non-increasing vector. Let $h : \mathbb{R}^m \to \mathbb{R}_{\geq 0}$ be a monotone, symmetric norm. Let $\varepsilon, \kappa > 0$. The expansion of v, $v^{exp} \in \mathbb{R}_{\geq 0}^m$, is given by $v_i^{exp} := v_i$ for $i \in POS$, and $v_i^{exp} = v_{prev(i)}$ for $i \in [m] \setminus POS$.*

(a) If $u_\ell^{\downarrow} \leq v_\ell$ for all $\ell \in POS$, then $Top_i(u) \leq Top_i(v^{exp})$ for all $i \in [m]$, and hence, $h(u) \leq h(v^{exp})$.

(b) If $v_\ell \leq (1+\varepsilon)u_\ell^{\downarrow} + \kappa$ for all $\ell \in POS$, then $Top_i(v^{exp}) \leq (1+\delta)(1+\varepsilon)\,Top_i(u) + i\kappa$ for all $i \in [m]$, and hence, $h(v^{exp}) \leq (1+\delta)(1+\varepsilon)h(u) + m\kappa \cdot h(1, 0, \ldots, 0)$.

Algorithm 13: Adaptive sampling algorithm for minimum norm k-clustering

Input: A minimum-norm k-clustering instance $(\mathcal{C}, d, k, f)$

A set $\mathcal{T}$ of non-increasing threshold vectors s.t. $\exists \vec{t} \in \mathcal{T}$ with $t^*_\ell \le \vec{t}_\ell \le (1+\varepsilon)t^*_\ell$ for all $\ell \in POS$

$\mathcal{S} \leftarrow \varnothing$
for $\vec{t} \in \mathcal{T}$ **do**
  $S_0 \leftarrow \varnothing$
  $\mathcal{L}_1(\vec{t}) \leftarrow POS$
  for $i = 1, \dots, \tau(k+\sqrt{k})$ **do**
    $\ell^*_i \leftarrow \max_{\ell \in \mathcal{L}_i(\vec{t})} \ell$
    Sample s_i with probability proportional to $(d(s_i, S_{i-1}) - \beta \vec{t}_{\ell^*_i})^+$
    Update $S_i \leftarrow S_{i-1} \cup \{s_i\}$
    Update $\mathcal{L}_{i+1}(\vec{t}) \leftarrow \left\{\ell \in POS : \mathrm{Top}_\ell(d(\mathcal{C}, S_i)) > \rho \cdot (1+\varepsilon) \cdot \mathrm{Top}_\ell(\vec{t}^{exp})\right\}$
  end
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{S_{\tau(k+\sqrt{k})}\}$
end
return $\arg\min_{S \in \mathcal{S}} f(d(\mathcal{C}, S))$

Theorem 7.0.10. *Let S be the set of centers opened by Algorithm 13 and let ρ be a constant that is strictly larger than 30. Then, there exists suitable parameters such that $|S| = O(k)$ and with constant probability, $\mathrm{Top}_\ell(d(\mathcal{C}, S)) \le \rho(1+\delta)^2(1+\varepsilon)^2 \cdot OPT$ with constant probability. Furthermore, this algorithm can be implemented in $O(\varepsilon^{-1} \cdot \max\{(\frac{n}{\varepsilon})^{O(1/\varepsilon)}, n^{1/\delta}\} \cdot k^2 \log n)$ time.*

Proof of Theorem 7.0.10. Let $\alpha, \beta, \gamma, \kappa$ be fixed constants satisfying $\alpha > 1$, $\gamma \ge \alpha + 1$, $\frac{1}{4}\left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right) \ge \frac{\kappa+\beta}{\rho}$, and $\kappa \ge \alpha + \beta + 1$, and define $\tau = \left(\left(1 - \frac{\max\{\beta,\gamma\}}{\rho}\right) \cdot \frac{\alpha-1}{2\kappa\alpha}\right)^{-1}$.

There exists a threshold vector $\tilde{t} \in \mathcal{T}$ such that $t^*_\ell \le \tilde{t}_\ell \le (1+\varepsilon)t^*_\ell$ for all $\ell \in POS$. Consider the iteration when $\tilde{t}$ is used as the guess for the threshold vector. In this iteration, if $\ell \notin \mathcal{L}_i(\tilde{t})$, $\mathrm{Top}_\ell(d(\mathcal{C}, S)) \le \rho(1+\varepsilon) \cdot \mathrm{Top}_\ell(\tilde{t}^{exp}) \le \rho \cdot (1+\delta)(1+\varepsilon)^2 \cdot \mathrm{Top}_\ell(d(\mathcal{C}, S^*))$, where the last inequality is due to Lemma 7.0.9. By Corollary 7.0.6, $\mathcal{L}_{\tau(k+\sqrt{k})}(\tilde{t}) = \varnothing$ with constant probability – so with constant probability, $\mathrm{Top}_\ell(d(\mathcal{C}, S)) \le \rho(1+\delta)(1+\varepsilon)^2 \cdot \mathrm{Top}_\ell(d(\mathcal{C}, S^*))$

for all $\ell \in [n]$, and hence by Theorem 7.0.3, $f(d(\mathcal{C}, S)) \leq \rho(1+\delta)^2(1+\varepsilon)^2 \cdot OPT$.

It remains to prove the bound on the running time of this algorithm. Given a candidate threshold vector $\vec{t}$, the innermost loop of Algorithm 13 runs in $O(k^2)$ time. Since, by Corollary 7.0.8, we can obtain a set $\mathcal{T} \subseteq \mathbb{R}_+^{POS}$ such that $|\mathcal{T}| = O(\log_{1+\epsilon} n \cdot \max\{(\frac{n}{\varepsilon})^{O(1/\varepsilon)}, n^{1/\delta}\})$ and $\mathcal{T}$ contains a valid threshold vector $\tilde{t}$ that is a good guess of t^*, the runtime of Algorithm 13 is $O(\varepsilon^{-1} \max\{(\frac{n}{\varepsilon})^{O(1/\varepsilon)}, n^{1/\delta}\} \cdot k^2 \log n)$. □

In particular, if we set $\alpha = 1.92, \beta = 2, \gamma = 2.92$, and $\kappa = 5.92$, we get a $(35(1+\varepsilon), 28(1+1/\sqrt{k}))$-bicriteria approximation, as stated by the following corollary.

Corollary 7.0.11. *If $\beta = 2$ and $\tau = 28$, Algorithm 13 yields a solution that opens $28(k+\sqrt{k})$ centers and has cost at most $35(1+\delta)^2(1+\varepsilon)^2 \cdot OPT$, with constant probability.*

Chapter 8

Conclusions and Future Work

In this thesis, we saw $O(1)$-distortion mechanisms for the k-median problem that use relatively few value queries per agent. We also saw low-distortion mechanisms for the ℓ-centrum (with a slightly worse query complexity); en route, we developed a simple sampling algorithm for the ℓ-centrum k-clustering problem, which we were able to extend to the minimum-norm setting as well. For the single-winner (1-median) problem, we gave a novel LP-duality based analysis framework that makes it easier to analyze the distortion of existing social choice functions. Using this framework, we gave simpler proofs of some known results. We also showed that this framework readily extends to randomized social choice functions.

There are a few interesting questions left open by this work. Perhaps the most obvious question is whether there exists $O(1)$-distortion mechanisms for the problems studied in this thesis that require fewer queries per agent. Another important question is whether one can give a lower bound on the number of queries that is required by any $O(1)$-distortion mechanism for the k-median problem. As the agents lie in a metric space, information regarding the distance between a pair of agents can be obtained without directly querying the agents, as queried edges can be used to infer bounds on the unqueried edges (via the triangle inequality). This highly correlated nature of the underlying distances makes it difficult to establish a lower bound on the number of queries required by an $O(1)$-distortion mechanism.

In this thesis, we have only studied the value query model, where we can ask an agent for the exact distance between herself and another agent. However, in some applications, computing the exact distance between a candidate and herself is difficult for the agent, and instead, it is easier for her to identify which candidates are at a distance of at most r from her location. We will refer to such queries as *ball queries*. The blackbox reductions we used (Algorithms 3 and 7) can in fact be implemented using $O(\log n)$ ball queries per agent. The difficulty lies in computing an initial estimate of OPT, as it is a non-trivial task to grasp the magnitude of the edge costs using relatively few ball queries. One could also consider other types of queries (e.g. the threshold queries used by Ma et. al [34], comparison queries used by Amanatidis et. al [4]), or other sources of limited cardinal information (e.g. aggregated information regarding voter passion as used by Abramowitz et. al [2]).

We also exclusively considered adaptive query models – that is, each query made by our mechanism depends on the answers to the previous queries. A pressing question is whether it is possible to design $O(1)$-distortion mechanisms that make a limited number of non-adaptive queries, i.e. queries that are determined by the instance $(\mathcal{C}, \sigma)$ alone, and not the answers to the previous queries.

In the single winner setting, a long-standing open question is whether there exists a randomized social choice function with distortion at most $3 - \varepsilon$ for some fixed constant $\varepsilon > 0$. It is also unclear whether the current lower bound of approximately 2.11 [22] is the best bound possible. Similar to how we derived a sufficient condition for 3-distortion candidacy in Chapter 3, it may also be possible to derive similar sufficient conditions for the randomized setting, using our analysis framework. It would be very interesting to see if one could bound the gap between (Q^{σ}_{ao}) and its relaxation (P^{σ}_{ao}).

References

[1] Ben Abramowitz and Elliot Anshelevich. Utilitarians without utilities: Maximizing social welfare for graph problems using only ordinal preferences. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 894–901, 2018.

[2] Ben Abramowitz, Elliot Anshelevich, and Wennan Zhu. Awareness of Voter Passion Greatly Improves the Distortion of Metric Social Choice. In *Web and Internet Economics - 15th International Conference, WINE 2019*, pages 3–16, 2019.

[3] Ankit Aggarwal, Amit Deshpande, and Ravi Kannan. Adaptive Sampling for k-Means Clustering. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 15–28. 2009.

[4] Georgios Amanatidis, Georgios Birmpas, Aris Filos-Ratsikas, and Alexandros A. Voudouris. Peeking behind the ordinal curtain: Improving distortion via cardinal queries. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020*, pages 1782–1789, 2020.

[5] Ioannis Anagnostides, Dimitris Fotakis, and Panagiotis Patsilinakos. Metric-Distortion Bounds under Limited Information. *arXiv:2107.02489 [cs]*, 2021.

[6] Elliot Anshelevich, Onkar Bhardwaj, Edith Elkind, John Postl, and Piotr Skowron. Approximating optimal social choice under metric preferences. *Artificial Intelligence*, pages 27–51, 2018.

[7] Elliot Anshelevich, Aris Filos-Ratsikas, Nisarg Shah, and Alexandros A. Voudouris. Distortion in Social Choice Problems: The First 15 Years and Beyond. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021*, pages 4294–4301, 2021.

[8] Elliot Anshelevich and John Postl. Randomized Social Choice Functions Under Metric Preferences. *Journal of Artificial Intelligence Research*, pages 797–827, 2017.

[9] Elliot Anshelevich and Shreyas Sekar. Blind, greedy, and random: Algorithms for matching and clustering using only ordinal information. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI 2016*, pages 390–396, 2016.

[10] Elliot Anshelevich and Wennan Zhu. Ordinal Approximation for Social Choice, Matching, and Facility Location Problems Given Candidate Positions. In *Web and Internet Economics - 14th International Conference, WINE 2018*, pages 3–20, 2018.

[11] Ali Aouad and Danny Segev. The ordered k-median problem: surrogate models and approximation algorithms. *Mathematical Programming*, pages 55–83, 2018.

[12] Surender Baswana and Sandeep Sen. A simple and linear time randomized algorithm for computing sparse spanners in weighted graphs. *Random Structures & Algorithms*, pages 532–563, 2007.

[13] Allan Borodin, Daniel Halpern, Mohamad Latifian, and Nisarg Shah. Distortion in voting with top-t preferences. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 22*, pages 116–122, 2022.

[14] Jarosław Byrka, Thomas Pensyl, Bartosz Rybicki, Aravind Srinivasan, and Khoa Trinh. An improved approximation for k-median and positive correlation in budgeted optimization. *ACM Trans. Algorithms*, pages 1–31, 2017.

[15] Jaroslaw Byrka, Krzysztof Sornat, and Joachim Spoerhase. Constant-factor approximation for ordered k-median. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 620–631, 2018.

[16] Ioannis Caragiannis, Aris Filos-Ratsikas, Swaprava Nath, and Alexandros Voudouris. Truthful mechanisms for ownership transfer with expert advice. *Mathematical Programming*, 2018.

[17] Ioannis Caragiannis and Ariel D. Procaccia. Voting almost maximizes social welfare despite limited communication. *Artificial Intelligence*, pages 1655–1671, 2011.

[18] Ioannis Caragiannis, Nisarg Shah, and Alexandros A. Voudouris. The metric distortion of multiwinner voting. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022*, pages 4900–4907, 2022.

[19] Deeparnab Chakrabarty and Chaitanya Swamy. Interpolating between k-Median and k-Center: Approximation Algorithms for Ordered k-Median. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, 2018.

[20] Deeparnab Chakrabarty and Chaitanya Swamy. Approximation algorithms for minimum norm and ordered optimization problems. In *Proceedings of the 51st ACM Symposium on Theory of Computing (STOC 2019)*, pages 126–137, 2019.

[21] Moses Charikar, Liadan O'Callaghan, and Rina Panigrahy. Better streaming algorithms for clustering problems. In *Proceedings of the Thirty-Fifth ACM Symposium on Theory of Computing (STOC 2003)*, pages 30–39, 2003.

[22] Moses Charikar and Prasanna Ramakrishnan. Metric Distortion Bounds for Randomized Social Choice. *arXiv:2111.03694 [cs]*, 2021.

[23] Xujin Chen, Minming Li, and Chenhao Wang. Favorite-candidate voting for eliminating the least popular candidate in a metric space. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020*, pages 1894–1901, 2020.

[24] Vasilis Gkatzelis, Daniel Halpern, and Nisarg Shah. Resolving the Optimal Metric Distortion Conjecture. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020*, pages 1427–1438, 2020.

[25] Ashish Goel, Reyna Hulett, and Anilesh K. Krishnaswamy. Relating Metric Distortion and Fairness of Social Choice Rules. *arXiv:1810.01092 [cs]*, 2018.

[26] Ashish Goel, Anilesh K. Krishnaswamy, and Kamesh Munagala. Metric Distortion of Social Choice Rules: Lower Bounds and Fairness Properties. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 287–304, 2017.

[27] Sharat Ibrahimpur and Chaitanya Swamy. Minimum-Norm Load Balancing Is (Almost) as Easy as Minimizing Makespan. In *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021*, pages 1–20, 2021.

[28] David Kempe. An Analysis Framework for Metric Voting based on LP Duality. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020*, pages 2079–2086, 2020.

[29] David Kempe. Communication, distortion, and randomness in metric voting. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020*, pages 2087–2094, 2020.

[30] Fatih Erdem Kizilkaya and David Kempe. Plurality Veto: A Simple Voting Rule Achieving Optimal Metric Distortion. *arXiv:2206.07098*, 2022.

[31] Amit Kumar and Jon Kleinberg. Fairness measures for resource allocation. *SIAM Journal on Computing*, pages 657–680, 2006.

[32] Gilbert Laporte, Stefan Nickel, and Francisco Saldanha da Gama. *Location Science*. Springer, 2015.

[33] Edo Liberty, Ram Sriharsha, and Maxim Sviridenko. An Algorithm for Online K-Means Clustering. In *2016 Proceedings of the Eighteenth Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 81–89, 2016.

[34] Thomas Ma, Vijay Menon, and Kate Larson. Improving Welfare in One-Sided Matchings using Simple Threshold Queries. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021*, pages 321–327, 2021.

[35] Adam Meyerson. Online facility location. pages 426–431, November 2001.

[36] Herve Moulin. Choosing from a tournament. *Social Choice and Welfare*, pages 271–291, 1986.

[37] Kamesh Munagala and Kangning Wang. Improved Metric Distortion for Deterministic Social Choice Rules. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 245–262, 2019.

[38] Stefan Nickel and Justo Puerto. *Location theory: a unified approach*. Springer Science & Business Media, 2006.

[39] Ariel D. Procaccia and Jeffrey S. Rosenschein. The distortion of cardinal preferences in voting. In *International Workshop on Cooperative Information Agents*, pages 317–331, 2006.

[40] Haripriya Pulyassary and Chaitanya Swamy. On the Randomized Metric Distortion Conjecture. *arXiv:2111.08698 [cs]*, 2021.

[41] Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems*. Cambridge University Press, 2009.

[42] Arie Tamir. The k-centrum multi-facility location problem. *Discrete Applied Mathematics*, pages 293–307, 2001.

APPENDICES

Appendix A

Expansion of (Best-Dist)

The expansion of (Best-Dist) is given below. We use $\mathcal{T}_{\ell A}$ to denote the set of triangles $i, j, k \in \mathcal{C} \cup A$ that contain ℓ candidates. More precisely, $\mathcal{T}_{0A}$ denotes triangles consisting of three agents, $\mathcal{T}_{1A}$ denotes triangles consisting of a single candidate and two agents, $\mathcal{T}_{2A} = \{(j, i_1, i_2) : j \in C, i_1, i_2 \in F, i_1 \neq i_2, i_1 \succeq_j i_2\}$, and $\mathcal{T}_{3A}$ denotes triangles consisting of three candidates.

$$\min \quad \gamma \tag{A.1}$$

$$\begin{aligned}\text{s.t.} \quad & \sum_{T=(i,j,\cdot)\in\mathcal{T}_{1A}} (-\alpha_{1,T}^{(1),o} + \alpha_{2,T}^{(1),o} - \alpha_{3,T}^{(1),o}) + \sum_{T=(i,\cdot,j)\in\mathcal{T}_{1A}} (-\alpha_{1,T}^{(1),o} - \alpha_{2,T}^{(1),o} + \alpha_{3,T}^{(1),o}) \\ & + \sum_{T=(j,\cdot,i)\in\mathcal{T}_{2A}} (\alpha_{1,T}^{(2),o} - \alpha_{2,T}^{(2),o}) + \sum_{T=(j,i,\cdot)\in\mathcal{T}_{2A}} (-\alpha_{1,T}^{(2),o} - \alpha_{2,T}^{(2),o}) + \beta_{j1}^{o}\mathbb{I}_{[i=\text{alt}_\sigma(j,1)]} \\ & + \sum_{r=1}^{n-2} (\beta_{j,r+1}^{o} - \beta_{j,r}^{o})\mathbb{I}_{[i=\text{alt}_\sigma(j,r+1)]} - \beta_{j,m-1}^{o}\mathbb{I}_{[i=\text{alt}_\sigma(j,m)]} \\ & + \gamma\mathbb{I}_{[i=o]} \geq q_i \qquad \forall j \in \mathcal{C}, \forall i, o \in A \end{aligned} \tag{A.2}$$

$$\begin{aligned} & \sum_{T=(i_1,i_2,\cdot)\in\mathcal{T}_{3A}} (\alpha_{1,T}^{(3),o} - \alpha_{2,T}^{(3),o} - \alpha_{3,T}^{(3),o}) + \sum_{T=(i_1,\cdot,i_2)\in\mathcal{T}_{3A}} (-\alpha_{1,T}^{(3),o} + \alpha_{2,T}^{(3),o} - \alpha_{3,T}^{(3),o}) \\ & + \sum_{T=(\cdot,i_1,i_2)\in\mathcal{T}_{3A}} (-\alpha_{1,T}^{(3),o} - \alpha_{2,T}^{(3),o} + \alpha_{3,T}^{(3),o}) \end{aligned} \tag{A.3}$$

$$+ \sum_{T\in\mathcal{T}_{2A}: i_1,i_2\in T} (\alpha_{2,T}^{(2),o} - \alpha_{1,T}^{(2),o}) \geq 0 \qquad \forall i_1, i_2, o \in A \tag{A.4}$$

$$\begin{aligned} & \sum_{(j_1,j_2,\cdot)\in\mathcal{T}_{0A}} (\alpha_{1,T}^{(0),o} - \alpha_{2,T}^{(0),o} - \alpha_{3,T}^{(0),o}) + \sum_{(j_1,\cdot,j_2)\in\mathcal{T}_{0A}} (-\alpha_{1,T}^{(0),o} + \alpha_{2,T}^{(0),o} - \alpha_{3,T}^{(0),o}) \\ & + \sum_{(\cdot,j_1,j_2)\in\mathcal{T}_{0A}} (-\alpha_{1,T}^{(0),o} - \alpha_{2,T}^{(0),o} + \alpha_{3,T}^{(0),o}) \end{aligned} \tag{A.5}$$

$$+ \sum_{(\cdot,j_1,j_2)\in\mathcal{T}_{1A}} (\alpha_{1,T}^{(1),o} - \alpha_{2,T}^{(1),o} - \alpha_{3,T}^{(1),o}) \geq 0 \qquad \forall j_1, j_2 \in \mathcal{C},\ \forall o \in A \tag{A.6}$$

$$\sum_{i\in A} q_i \geq 1 \tag{A.7}$$

$$\alpha^{(0),o}, \alpha^{(1),o}, \alpha^{(2),o}, \alpha^{(3),o}, \beta^{o}, \gamma, q \geq 0 \qquad \forall o \in A \tag{A.8}$$

Appendix B

Low distortion algorithms for other social cost minimization problems

In the earlier chapters, we have exclusively studied the k-median/k-winner selection problem. However, the notion of distortion can also be extended to purely-ordinal algorithms and mechanisms for other social cost minimization problems as well. To be precise, in this class of optimization problems, a solution induces a cost for each agent, and the objective is to minimize the total sum of costs incurred by the agents.

Definition B.0.1 (Distortion of algorithms and mechanisms for arbitrary SCM problems)**.** Consider a social cost minimization problem and let $\mathcal{A}$ and $\mathcal{M} = (\mathcal{Q}, f, k)$ be a purely-ordinal algorithm and mechanism respectively. Let $\chi(\mathcal{C}, \sigma)$ be the set of feasible solutions for a given instance. We define the distortion of $\mathcal{A}$ and $\mathcal{M}$ to be

$$\text{distortion}(\mathcal{A}) = \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} cost(j, \mathcal{A}(\sigma); d)}{\min_{S \in \chi(\mathcal{C}, \sigma)} \sum_{j \in \mathcal{C}} cost(j, S; d)}$$

$$\text{distortion}(\mathcal{M}) = \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\sum_{j \in \mathcal{C}} cost(j, \mathcal{M}(\sigma|d); d)}{\min_{S \in \chi(\mathcal{C}, \sigma)} \sum_{j \in \mathcal{C}} cost(j, S; d)}$$

where $d \triangleleft \sigma$ denotes that d is consistent with σ.

In this chapter we design low-distortion mechanisms and purely-ordinal algorithms for other social cost minimization problems.

B.1 Minimum spanning tree

We have already seen (in Remark 4.0.2) that the distortion of any purely-ordinal algorithm for the minimum cost k-Forest problem is unbounded, for $k \geq 3$. Given this result, a natural question is whether there exists a purely-ordinal algorithm for minimum spanning tree problem that has bounded distortion.

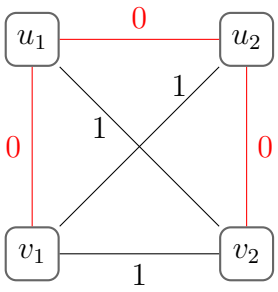
Lower bound for the general (non-metric) case

Without the metric assumption, the distortion of any purely-ordinal MST algorithm is unbounded. To see this, consider the following example (similar to [1, Example 1]).

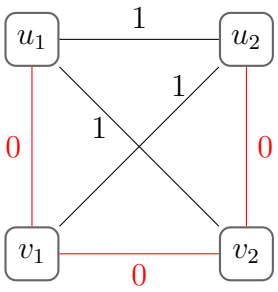
Example B.1. Let $(\mathcal{C}, \sigma)$ be an instance with $\mathcal{C} = \{u_1, u_2, v_1, v_2\}$. The preference profile σ is

$$u_1 : v_1 \succeq u_2 \succeq v_2 \quad u_2 : v_2 \succeq u_1 \succeq v_1 \quad v_1 : u_1 \succeq v_2 \succeq u_2 \quad v_2 : u_2 \succeq v_1 \succeq u_1$$

The following figure shows two cost vectors, $c^{(1)}$ and $c^{(2)}$, which are consistent with σ. The optimal solution under each cost vector is depicted in red.



(a) Cost vector $c^{(1)}$



(b) Cost vector $c^{(2)}$

Figure B.1: An MST instance with unbounded distortion

As shown in Figure B.1, for both $c^{(1)}$ and $c^{(2)}$, the cost of an optimal solution is 0, and any other spanning tree will have strictly positive cost. Since a purely ordinal algorithm cannot differentiate between $c^{(1)}$ and $c^{(2)}$, the distortion of any purely ordinal algorithm for MST is unbounded.

A purely-ordinal algorithm for the metric case

For the metric MST problem, any spanning tree is an $(n-1)$-approximate solution, so a purely-ordinal algorithm is guaranteed to have bounded distortion. We show that a purely-ordinal analogue of Kruskal's algorithm has a slightly better bound (namely $\frac{n}{2}$), though we conjecture that this bound can be significantly improved for this algorithm.

If $d(i,j)$ was known for all $i,j \in \mathcal{C}$, an MST could be computed easily using Kruskal's greedy algorithm: repeatedly choose the least-cost edge whose endpoints are in different components, until we have a spanning tree. Of course, we do not know $d(i,j)$ for all $i,j \in \mathcal{C}$. However, as observed by Anshelevich and Sekar [9], given the preference profile σ, it is possible to obtain a set of edges that is guaranteed to contain a maximum/minimum-cost edge. Anshelevich and Sekar introduced the notion of *undominated edges* in the context of a maximization problem. As we are interested in a minimization problem, we will work with the following modified definition.

Definition B.1.1 (Undominated edge)**.** An edge $\{x,y\} \in E'$ is undominated with respect to E' if $d(x,y) \leq d(x,z)$ for all $\{x,z\} \in E'$ and $d(x,y) \leq d(y,z)$ for all $\{y,z\} \in E'$.

Observe that any minimum-cost edge of E' is an undominated edge with respect to E'. One can obtain a set of undominated edges by using Anshelevich and Sekar's procedure [9].

Algorithm 14: Procedure for finding undominated edges [9]

```
U ← ∅
for x ∈ C do
    Let y be the agent closest to x such that {x, y} ∈ E'. Let z be the agent
     closest to y such that {y, z} ∈ E'
    if x = z then
        U ← U ∪ {x, y}
    end
    Repeat this process with z. Either, at some point we will get a single edge
     (a, b), or we will get a cycle (in which case add all edges of the cycle to U.)
end
return U
```

The main idea of Algorithm 15 is to run a greedy algorithm where, instead of choosing the minimum cost edge, we choose an undominated edge; these edges, by definition, should be of relatively low cost.

Algorithm 15: Algorithm for MST

$F \leftarrow \varnothing$
$E' \leftarrow \{\{i,j\} : i,j \in \mathcal{C}\}$
$U = \{\text{undominated edges in } E'\}$
while $(\mathcal{C}, F)$ *has more than* 1 *components* **do**
  Choose $e^* \in U$
  $F \leftarrow F \cup \{e^*\}$
  $E' \leftarrow E' \setminus \bigcup_{C_i \text{ component of } F} E(C_i)$
  $U \leftarrow \{\text{undominated edges in } E'\}$
end
return F

Theorem B.1.2. *Algorithm 15 is a $\frac{n}{2}$-distortion algorithm for (metric) MST.*

Proof. Consider an arbitrary instance $(\mathcal{C}, \sigma)$, and assume distinct edge costs (this assumption is without loss of generality, as we can fix a tie-breaking rule on the edges). Let T be the solution returned by Algorithm 15, $(\mathcal{C}, d^*)$ be the true underlying metric, and T^* be the MST with respect to $(\mathcal{C}, d^*)$. Since T is a spanning tree, $|\delta(j) \cap T| = 1$ for every $j \in \mathcal{C}$. Furthermore, T only consists of undominated edges – and hence, the minimum-cost edge incident to each vertex must be added. Since, for all $j \in \mathcal{C}$, the minimum cost edge in $\delta(j)$ must be in T^*, $|T^* \cap T| \geq \left\lfloor \frac{n}{2} \right\rfloor$. For $e = uv \in T \setminus T^*$, there exists a u-v path P_{uv} in T^*, and since the edge costs satisfy the triangle inequality, $d^*(u,v) \leq \sum_{e' \in P_{uv}} d^*(e') \leq \text{OPT}$. Thus, $\sum_{e \in T} d^*(e) \leq |T \setminus T^*| \cdot \text{OPT} \leq \left(n - 1 - \left\lfloor \frac{n}{2} \right\rfloor\right) \cdot \text{OPT} \leq \frac{n}{2} \cdot \text{OPT}$. □

However, this bound is unlikely to be tight. We conjecture that the distortion of Algorithm 15 is much lower than $\frac{n}{2}$.

Conjecture B.1.3. Algorithm 15 is a 2-distortion algorithm for MST.

B.2 Other social cost minimization problems

Recall that we can extend the notion of mechanisms and distortion to arbitrary social cost minimization problems. The blackbox reduction given in Chapter 3 can be used to obtain an $O(1)$-distortion mechanism using $O(\log^2 n)$ value queries per agent for any social-cost minimization problem provided that (1) there exists an $O(1)$ polynomial-time approximation algorithm for the problem, (2) one can compute B, a good estimate of OPT using $O(\log^2 n)$ value queries per agent, and (3) given this estimate B, Claim 5.1.4 holds.

Depending on the problem, it may not be possible to compute an estimate using so few value queries. We suggest a different approach using spanners that has a worse query complexity than Algorithm 3, but avoids this bootstrapping issue altogether. A *t-spanner* of $G = (V, E)$ is a (sparse) subgraph (V, E_S), $E_S \subseteq E$ such that for any $u, v \in V$, if $d(u, v)$ is the distance between u and v in G and P_{uv} is the shortest u, v-path in (V, E_S), $d(u, v) \leq \sum_{e \in P_{uv}} d(e) \leq 3 \cdot d(u, v)$. Baswana and Sen [12] gave a simple randomized algorithm for computing a 3-spanner.

Algorithm 16: Randomized 3-spanner algorithm [12]

$E_S \leftarrow \varnothing$
$\mathcal{R}$: random sample of $\mathcal{C}$ chosen by picking each vertex independently with probability $\frac{1}{\sqrt{n}}$.
$C_1, \ldots, C_k$: clusters induced by $\mathcal{R}$ (by assigning $\mathcal{C} \setminus \mathcal{R}$ to the closest center in $\mathcal{R}$)
for $j \in \mathcal{C} \setminus \mathcal{R}$ **do**
    $E_S \leftarrow E_S \cup \{(j, i) : i \succeq_j top_{\mathcal{R}}(j)\}$
end
for $j \in \mathcal{C}$ **do**
    for $C_i : j \notin C_i$ **do**
        $E_S \leftarrow E_S \cup \{(j, top_{C_i}(j))\}$
    end
end
return E_S

Theorem B.2.1 (Baswana and Sen [12])**.** *E_S is a 3-spanner, and for any $j \in \mathcal{C}$, $\mathbb{E}[|E_S|] = 2n\sqrt{n}$.*

Theorem B.2.2. *For any social cost minimization problem where the objective is a monotone, subadditive function of the (metric) distances between agents, there exists a $O(1)$-distortion (randomized) mechanism that requires $O(\sqrt{n})$ value queries per agent (amortized).*

Proof. Let E_S be the set of edges returned by Algorithm 16. Assume that $|E_S| \leq 4n\sqrt{n}$ (notice that by Markov's inequality, if Algorithm 16 is repeated $O(1)$ times, we will obtain E_S of size at most $4n\sqrt{n}$ with constant probability). For each $\{u, v\} \in E_S$, we query u for the value of $d(u, v)$. On average, this requires a total of at most $4\sqrt{n}$ value queries per agent (amortized). We can define a simulated metric $\tilde{d}$ where $\tilde{d}(u, v)$ is the shortest-path distance between u and v in $(\mathcal{C}, E_S)$, for any $u, v \in \mathcal{C}$. Moreover, as E_S is a 3-spanner, if d is the true underlying metric, $d(u, v) \leq \tilde{d}(u, v) \leq 3d(u, v)$ for any $u, v \in \mathcal{C}$. So, for any minimization problem where the objective is a monotone, subadditive function of d, we can work with $\tilde{d}$ instead of d, losing only a factor of 3. Thus, for such problems, we can obtain an $O(1)$-distortion (randomized) mechanism that requires $O(\sqrt{n})$ queries per agent (amortized). □