

Constant-Factor Distortion Mechanisms for k -Committee Election

Hari Priya Pulyassary¹, Chaitanya Swamy²

¹School of ORIE, Cornell University, Ithaca, NY 14853, USA

²Dept. of Combinatorics & Optimization, University of Waterloo, Waterloo, ON N2L 3G1, CANADA
hp297@cornell.edu, cswamy@uwaterloo.ca

Abstract

In the k -committee election problem, we wish to aggregate the preferences of n agents over a set of alternatives and select a committee of k alternatives that minimizes the cost incurred by the agents. While we typically assume that agent preferences are captured by a cardinal utility function, in many contexts we only have access to ordinal information, namely the agents' rankings over the outcomes. As preference rankings are not as expressive as cardinal utilities, a loss of efficiency is inevitable, and is quantified by the notion of *distortion*.

We study the problem of electing a k -committee that minimizes the sum of the ℓ -largest costs incurred by the agents, when agents and candidates are embedded in a metric space. This problem is called the ℓ -centrum problem and captures both the utilitarian and egalitarian objectives. When $k \geq 2$, it is not possible to compute a bounded-distortion committee using purely ordinal information. We develop the first algorithms (that we call mechanisms) for the ℓ -centrum problem (when $k \geq 2$), which achieve $O(1)$ -distortion while eliciting only a very limited amount of cardinal information via value queries. We obtain two types of query-complexity guarantees: $O(\log k \log n)$ queries *per agent*, and $O(k^2 \log^2 n)$ queries *in total* (while achieving $O(1)$ -distortion in both cases). En route, we give a simple adaptive-sampling algorithm for the ℓ -centrum k -clustering problem.

1 Introduction

In many applications, 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' 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. 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 (2006) introduced

the notion of *distortion* to quantify the worst-case efficiency loss for a given social choice function.

Much of the prior work has primarily considered the *utilitarian* objective, which minimizes the sum of individual costs incurred by the agents. However, this utilitarian objective may not always be the appropriate choice. For instance, in some settings (e.g. where fairness is an important consideration), we may instead wish to consider an *egalitarian* objective and minimize the *maximum* cost incurred by any agent. Both objectives are special cases of the Top_ℓ objective, which minimizes the sum of the ℓ largest costs incurred by agents: clearly, when $\ell = 1$ and $\ell = n$, we recover the egalitarian and utilitarian objectives respectively.

In this work, we study the k -committee election problem, wherein each agent has a preference ordering over the set of candidates and we wish to elect a committee of k candidates, so as to minimize the Top_ℓ -cost. An instance of this problem (\mathcal{C}, A, σ) consists of a set of n agents or voters \mathcal{C} , a set of m alternatives or candidates A , and a preference profile (a tuple giving the preference ordering over A , for each agent), σ . In line with prior work, we consider the *metric setting*, wherein agents and candidates correspond to points in a metric space specified by a distance function $d : \mathcal{C} \times A \rightarrow \mathbb{R}_{\geq 0}$ satisfying the “triangle” inequality: for any $i, j \in \mathcal{C}$ and $a, b \in A$, we have $d(i, a) \leq d(i, b) + d(j, b) + d(j, a)$. We slightly abuse notation and use d to also denote the resulting metric. This assumption models many applications, including those where agents prefer alternatives that are ideologically similar to them: here $d(i, a)$ can be interpreted as the *ideological distance* between agent i and candidate a . As the preference profile σ arises from the distance function d , it must be that d is *consistent* with σ , denoted $d \triangleleft \sigma$: that is, for any $i \in \mathcal{C}$ and $a, b \in A$, if i prefers a over b , denoted $a \succeq_i b$, then $d(i, a) \leq d(i, b)$.

A social choice function (SCF) f for k -committee election maps a preference profile σ to a set in $A^k := \{S \subseteq A : |S| \leq k\}$. The cost incurred by an agent i , when a set S of candidates is chosen, is given by $d(i, S) := \min_{a \in S} d(i, a)$, i.e., the distance to the closest alternative in S . Since f does not know the cardinal information, one would not expect f to output the best solution for the given metric d , and the *distortion* (Procaccia and Rosenschein 2006) of f quantifies the worst-case loss in solution quality that can occur due to the fact that f does not have cardinal information.

More precisely, $\text{distortion}(f)$ is the worst case ratio (over all instances) of the cost of the solution output by f over the optimal cost; formally

$$\text{distortion}(f) = \sup_{\sigma} \sup_{d \sim \sigma} \frac{\text{Top}_{\ell}(d(\mathcal{C}, f(\sigma)))}{\min_{S \in A^k} \text{Top}_{\ell}(d(\mathcal{C}, S))},$$

where $d(\mathcal{C}, S)$ denotes the vector $\{d(i, S)\}_{i \in S}$ of agents' costs for $S \in A^k$, and $\text{Top}_{\ell}(d(\mathcal{C}, S))$ is the Top_{ℓ} -cost of this vector. Throughout, we use “algorithm” to refer to a procedure whose input includes complete cardinal information, i.e., the metric d , and use the term “mechanism” when the input includes only ordinal information given by the preference profile σ .

1.1 Our Contributions

We initiate the study of low-distortion mechanisms for k -committee election under the Top_{ℓ} objective. The underlying problem can be equivalently viewed as a k -clustering problem (clustering agents/points around k alternatives/centers), and we sometimes use the clustering-terminology, ℓ -centrum problem, to refer to this problem. As noted earlier, ℓ -centrum is a very versatile model, which generalizes, and interpolates between, the classical and extensively-studied k -center ($\ell = 1$) and k -median ($\ell = |\mathcal{C}| = n$) problems. Even for k -median, for any $k > 1$, it is *impossible* to obtain low-distortion mechanisms using just ordinal information; the distortion can be: (a) $\Omega(n)$ when $k = 2$ (Anshelevich and Zhu 2018), and (b) unbounded when $k > 2$ (Theorem 3.1). In light of this, a natural question that arises is: *can one achieve meaningful distortion bounds (for ℓ -centrum) by eliciting a small amount of cardinal information?*

We answer this question *affirmatively*. One of the simplest ways of obtaining cardinal information, which was also considered in some recent work on k -committee election (Burkhardt et al. 2024), is via a *value query*, wherein we query $d(i, j)$ for an agent-alternative pair (i, j) . Our chief contribution is to develop *constant-factor* distortion mechanisms for the ℓ -centrum problem using a very limited number of value queries.

We consider two ways of measuring query-complexity: (1) *per-agent* query complexity, which measures the maximum number of queries that any single agent is asked; and (2) *total* (or *average*) query complexity, wherein we bound the total number of queries elicited from the entire agent population. We devise mechanisms that achieve $O(1)$ distortion and obtain strong bounds under both query-complexity measures. We focus on the setting $A = \mathcal{C}$, though some of our results apply more generally (see Table 1). We obtain per-agent query-complexity bounds of $O(\log k \log n)$ and $\tilde{O}(k \cdot \log(\min\{\ell, n/\ell\}))$ (Mechanisms 2 and 3 respectively), where the $\tilde{O}(\cdot)$ notation suppresses $O(\log \log k)$ factors. Observe that the latter bound is *independent of n* , for any fixed ℓ as also for large ℓ (including the case $\ell = n$); in particular, for any fixed k and ℓ , we only make a *constant* number of queries per agent. The algorithmic idea leading to the latter bound is fairly robust, and we show that it can be implemented to also yield a total query-complexity bound

Per-agent queries	$O(\log k \log n)$	Mechanism 2
	$\tilde{O}(k \log(\min\{\ell, n/\ell\}))$	Mechanism 3
Total queries	$O(k^2 \log \ell \log^2 n)$	

Table 1: Summary of our results, in the setting $A = \mathcal{C}$. All mechanisms achieve $O(1)$ distortion. Mechanisms 2 and 3 can be generalized to the $A \neq \mathcal{C}$ setting to yield the same guarantees. We describe this, and our total-query-complexity mechanism in the full version.

of $O(k^2 \log \ell \log^2 n)$; observe that this implies that the *average* query complexity goes down to 0 as n grows!

Our mechanisms are randomized and achieve $O(1)$ -distortion with constant success probability.¹ They can be modified to achieve $O(1)$ -distortion *in expectation* with the same expected query-complexity bounds; this is discussed in the full version of the paper.

To our knowledge, these are the *first* results establishing distortion upper bounds for Top_{ℓ} k -committee election for $k > 1$. Some of these results were obtained in a preliminary form in Pulyassary (2021). Our results partially answer an open question posed in (Burkhardt et al. 2024) of obtaining small distortion for norm-based k -clustering objectives. While they consider a separate generalization of k -median, it is worth noting that for k -median, we obtain *significantly improved guarantees* compared to (Burkhardt et al. 2024): we obtain a true approximation, as opposed to bicriteria solutions, utilizing much fewer total number of queries, $O(k^2 \log^3 n)$, as opposed to $O(k^4 \log^5 n)$.

Technical contributions and overview. We focus on the $A = \mathcal{C}$ setting; in Section 4, we very briefly discuss extensions to the case $A \neq \mathcal{C}$. Table 1 summarizes our main results. Due to space limitations, we defer many of the details to the full version of the paper.

Our mechanisms consist of two chief ingredients. First, we compute a coarse estimate that approximates the optimal ℓ -centrum value, $OPT = OPT_{\ell}$, within $\text{poly}(n)$ factors. We actually estimate the optimal k -center or k -median value, which suffices, since all OPT_{ℓ} values are within a factor of n of each other: for $r \leq \ell$, we have $OPT_r \leq OPT_{\ell} \leq \frac{\ell}{r} \cdot OPT_r$. We utilize different methods for this, which differ in terms of their query-complexity bounds and the approximation quality of the estimate returned. We briefly discuss these methods below, and state the guarantees obtained.

(a) *Boruvka mechanism:* We use Boruvka’s algorithm for MSTs to find a *minimum-cost k -forest*, where a k -forest is a graph with k components. This procedure, which we refer to as the Boruvka mechanism, runs in $O(\log n)$ iterations, where each iteration uses at most 1 query per agent and merges every component with its “closest neighbor.”

¹We cannot detect if failure occurs, i.e., the distortion bound is not met, but we can boost the success probability by repetition, since we can evaluate the cost of a solution using one query per agent and at most $k\ell$ queries in total.

Theorem 1.1. *The Boruvka mechanism has $O(\log n)$ per-agent query complexity and returns an estimate B such that $OPT \leq B \leq n^2 \cdot OPT$.*

(b) *k -center and k -median mechanisms:* Here, we use certain approximation algorithms for k -center and k -median to obtain our estimate. These have the benefit that their query complexity is independent of n . For small ℓ , we use the well-known 2-approximate k -center algorithm (Gonzalez 1985). As observed by (Burkhardt et al. 2024), this can be implemented using $O(k)$ per-agent queries and $O(k^2)$ total number of queries. For large ℓ , we use k -means++ (Arthur and Vassilvitskii 2007), a randomized $O(\log k)$ -approximation algorithm for k -median that utilizes an elegant adaptive-sampling approach. *Adaptive sampling* is actually a core-algorithmic idea underlying some of our mechanisms (see below) that we adapt to directly handle the Top_ℓ -objective and obtain good total-query complexity, but a vanilla implementation easily yields $O(k)$ per-agent complexity.

Theorem 1.2. *In polynomial time, we can compute:*

- (a) *an estimate B_1 such that $OPT_\ell \leq B_1 \leq 2\ell \cdot OPT_\ell$ using $O(k)$ per-agent queries and $O(k^2)$ queries in all;*
- (b) *an estimate B_n such that $OPT_\ell \leq B_n \leq (8 \ln k + 4) \cdot \frac{n}{\ell} \cdot OPT_\ell$ holds with probability at least $1/2$ using $O(k)$ queries per agent.*

Second, and this is our chief technical contribution, we show how to leverage these estimates of OPT in combination with algorithmic ideas developed in the cardinal setting, to obtain mechanisms with $O(1)$ distortion and low query complexity. We develop two core algorithmic ideas.

1. Black-box reduction (Section 3.1). We present a simple, yet quite versatile reduction that *transforms the ordinal problem to the cardinal ℓ -centrum problem* (i.e., where we know the metric) using $\text{polylog}(n)$ value queries while incurring an $O(1)$ -factor loss in solution quality. We can then utilize any $O(1)$ -approximation algorithm for cardinal ℓ -centrum in a *black-box fashion* to obtain $O(1)$ distortion.

The reduction proceeds by approximating the true metric d^* by a sufficiently-close metric \tilde{d} ; see Mechanism 1 and Remark 3.2. Given an estimate $B \in [OPT, \alpha \cdot OPT]$, we consider each agent i . Roughly speaking, we partition $[\frac{\varepsilon B}{\alpha n}, B]$ into intervals $(\zeta, (1 + \varepsilon)\zeta]$, where the ζ values increase by a $(1 + \varepsilon)$ -factor. For each value ζ , we can use binary search on i 's preference relation to find all points a for which $d^*(i, a) \leq (1 + \varepsilon)\zeta$. This entire procedure uses $O(\log^2 n)$ value queries from i . Now we simply find any metric \tilde{d} that is consistent with this information, i.e., satisfies $\tilde{d}(i, a) \in (\zeta, (1 + \varepsilon)\zeta]$ whenever $d^*(i, a) \in (\zeta, (1 + \varepsilon)\zeta]$ and $\tilde{d}(i, a) \leq \frac{\varepsilon B}{\alpha n}$ whenever $d^*(i, a) \leq \frac{\varepsilon B}{\alpha n}$. It is not hard to argue that every solution has roughly the same cost under the \tilde{d} and d^* metrics; hence, we can work with the metric \tilde{d} !

To improve this to $O(\log k \log n)$ per-agent query complexity, we combine the above with a *sparsification* idea. We move to an instance with $O(k)$ distinct *weighted* points, losing an $O(1)$ -factor. Running the black-box reduction on this weighted instance now only requires $O(\log k \log n)$ queries per agent, since for each ζ value, we only need to use bi-

nary search over k points. We obtain the sparse instance by computing a bicriteria solution for ℓ -centrum that opens $O(k)$ centers and achieves $O(1)$ -approximation. We show that “moving” each point to its nearest center in this solution yields the desired sparse instance. We adapt the algorithm of (Meyerson 2001) for facility location to the ℓ -centrum setting (Algorithm 1), and show that by suitably using our estimate B , we can obtain the desired bicriteria solution. Mechanism 2 describes the combined mechanism.

2. Adaptive sampling for Top_ℓ -objective (Section 3.2).

We obtain per-agent query complexity that is independent of n , and total query-complexity bounds, by exploiting an elegant random-sampling approach called adaptive sampling due to (Aggarwal, Deshpande, and Kannan 2009) (see also (Arthur and Vassilvitskii 2007; Ostrovsky et al. 2012)), which yields good bicriteria solutions for k -median. In adaptive sampling, we pick a random point to add to the current center-set S choosing point i with probability proportional to $d(i, S)$, and we do this for $O(k)$ iterations. Observe that one value query to each agent i suffices to calculate $d(i, S)$, so this procedure uses $O(k)$ per-agent queries. The above approach does not directly work for ℓ -centrum. But we show that, by capitalizing on an insight of (Chakrabarty and Swamy 2019) that enables us to (roughly speaking) cast the Top_ℓ -objective as a k -median objective (see Claim 2.3), we can suitably modify the way the next center is sampled and adapt the approach to handle the ℓ -centrum problem (Algorithm 2).² We need to run this modified adaptive sampling $\tilde{O}(\log(\min\{\ell, n/\ell\}))$ times, using information gleaned from the estimates of OPT returned by Theorem 1.2, so this yields $\tilde{O}(k \log(\min\{\ell, n/\ell\}))$ per-agent query complexity (Mechanism 3).

To obtain the total query bound stated in Table 1, we execute adaptive sampling slightly differently. Instead of querying agents outside of S , we query agents in S , and we compute the $d(\mathcal{C}, S)$ cost-vector approximately. As in the black-box reduction, we consider geometrically-increasing distance thresholds within a $\text{poly}(n)$ -bounded range, and for each threshold ζ , we compute the ring of points $a \in \mathcal{C}$ for which $d(a, S) \in (\zeta, (1 + \varepsilon)\zeta]$. As before, this can be computed via binary search on j 's preference relation for each $j \in S$, so this takes $O(|S| \log^2 n)$ queries in total. Now, we can treat all points within a ring as having roughly the same $d(a, S)$ value, so we can approximately implement adaptive sampling by choosing a ring with the appropriate probability and then a uniform point within the ring. This yields the desired total query complexity.

1.2 Related Work

Distortion was first introduced and studied by Procaccia and Rosenschein (2006). Subsequent works (Anshelevich et al. 2018; Munagala and Wang 2019) studied the distortion of SCFs for *single-winner elections* in the metric setting and

²In fact, we can extend adaptive sampling to handle the general *minimum-norm k -clustering* problem (Chakrabarty and Swamy 2019); see Pulyassary (2021). The query complexity blows up prohibitively, but this is of interest in the cardinal setting.

conjectured that there exists a deterministic SCF with distortion of at most 3. This conjecture was ultimately resolved by Gkatzelis, Halpern, and Shah (2020), who gave a deterministic 3-distortion social choice function.

Furthermore, a series of work (Anshelevich and Postl 2017; Kempe 2020; Gkatzelis, Halpern, and Shah 2020) culminating in the recent 2.74-distortion (randomized) SCF by Charikar et al. (2024) showed that randomized SCFs can achieve strictly better distortion bounds. Here, a long-standing conjecture was that there exists a randomized SCF that achieves a distortion of 2; this conjecture was refuted independently by Charikar and Ramakrishnan (2022) and Pulyassary and Swamy (2021). This latter work also gave an LP to find an *instance-wise optimal randomized SCF*; i.e., the LP computes, for a given instance, the randomized SCF with smallest distortion.

We study k -committee election for $k > 1$. Committee election problems have been well-studied by the social choice community (see, for instance, Faliszewski et al. (2017) and references therein). Low-distortion algorithms of variants of the committee-election problem have been studied in the social-welfare-maximization setting (Borodin et al. 2022) and social-cost-minimization setting (Goel, Hulett, and Krishnaswamy 2018; Chen, Li, and Wang 2020; Caragiannis, Shah, and Voudouris 2022), however, these models are quite different from the one we consider.

In stark contrast to single-winner elections, Caragiannis, Shah, and Voudouris (2022) showed that the distortion of any k -committee election algorithm is unbounded in the cost-minimization setting (for $k > 2$). In light of this result, a natural question to ask is whether eliciting a small amount of additional cardinal information from the agents can yield better algorithms. This has been studied for single-winner elections (Amanatidis et al. 2020; Abramowitz, Anshelevich, and Zhu 2019), as well as other social choice problems, including matchings (Amanatidis et al. 2022; Anari, Charikar, and Ramakrishnan 2023). In the cost-minimization setting, when $A = C$, Burkhardt et al. (2024) present $O(1)$ -distortion mechanisms for k -committee election under the (k, z) -clustering objective, wherein one seeks to minimize $\sum_{j \in C} (d(j, S))^z$. A special case of this problem, when $z \rightarrow \infty$, is the k -center problem, wherein one minimizes the maximum induced assignment cost. For the k -center problem, Burkhardt et al. (2024) give a 2-distortion algorithm requiring a total of $O(k^2)$ value queries. They also give $O(1)$ -distortion mechanisms for the general (k, z) -clustering problem; these are *bicriteria* mechanisms, and consequently select a set of candidates of cardinality larger than k . Finally, as noted earlier, a preliminary version of these results was obtained in Pulyassary (2021).

2 Preliminaries

Recall that C is a set of n agents or voters, and A is a set of m alternatives or candidates. For $i \in C$, and $a, b \in A$, we say that $a \succeq_i b$ if agent i prefers candidate a to b . Each agent i 's preference relation \succeq_i induces a total order on A . We denote the top choice of $i \in C$ as $top(i, \succeq_i)$, or just $top(i)$ when \succeq_i is clear from the context. Similarly, we denote the

top choice of $i \in C$ 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 \succeq be the collection of all total orders on A . A preference profile is a tuple $\sigma = (\succeq_1, \dots, \succeq_n) \in \succeq^n$. As mentioned earlier, we consider the metric setting, where agents and candidates are located in a metric space specified by a distance function $d : C \times A \mapsto \mathbb{R}_{\geq 0}$ that satisfies the triangle inequality and is consistent with σ , denoted $d \triangleleft \sigma$: for any $i \in C$, and $a, b \in A$, if $a \succeq_i b$, then $d(i, a) \leq d(i, b)$.

The solution-space of the k -committee election problem is the collection of subsets of A of size at most k , denoted A^k . Any $S \in A^k$ induces a cost vector $d(C, S) := \{d(i, S)\}_{i \in C}$, where $d(i, S) := \min_{a \in S} d(i, a)$ is the cost incurred by i . The Top_ℓ -cost of a vector $v \in \mathbb{R}_{\geq 0}^n$ is the sum of the ℓ largest entries of v : $Top_\ell(v) = \sum_{i=1}^\ell v_i^\downarrow$, where v^\downarrow is the vector v with entries sorted in non-increasing order.

We consider k -committee election under the Top_ℓ objective, i.e., the cost of a solution $S \in A^k$ is $Top_\ell(d(C, S))$; we often refer to this as the ℓ -centrum problem. The special cases where $\ell = 1$ and $\ell = n$ correspond to the classical k -center and k -median problems respectively. While ℓ -centrum has been studied in the setting where the metric d is given, our focus is on devising mechanisms given the ordinal information specified by σ . In the absence of cardinal information, it is inevitable that any social choice function $f : \succeq \mapsto A^k$ must incur some loss in solution quality. This loss is quantified using the notion of distortion.

Definition 2.1. Let $f : \succeq \rightarrow A^k$ be a social choice function for k -committee election. The *distortion of f* is defined as

$$\text{distortion}(f) := \sup_{\sigma} \sup_{d \triangleleft \sigma} \frac{\text{Top}_\ell(d(C, f(\sigma)))}{\min_{S \in A^k} \text{Top}_\ell(d(C, S))}.$$

We seek mechanisms with low distortion, but as noted earlier, this is impossible given *only* ordinal information, and we must use some cardinal information to achieve any bounded distortion. While different query models for eliciting cardinal information have enjoyed varying levels of success for *social-welfare maximization* problems (Amanatidis et al. 2020; Ma, Menon, and Larson 2021), much less is known for the cost-minimization setting. One simple and very natural query is a *value query* (also used by Burkhardt et al. (2024)), where we query agent i for the distance $d(i, a)$ between itself and alternative a . We consider mechanisms that utilize (a limited number of) value queries, and extend the notion of distortion accordingly.

Definition 2.2. A *mechanism \mathcal{M}* for k -committee election takes as input a preference profile σ , can adaptively make value queries, and outputs some solution $S \in A^k$. The output of \mathcal{M} can depend on d , but only via the answers of the value queries made by it. We use $\mathcal{M}(\sigma|d)$ to denote the output of \mathcal{M} on input σ when the underlying metric is d .

The *distortion of \mathcal{M}* is defined as:

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

Handling the Top_ℓ objective. The Top_ℓ objective can be difficult to work with due to its non-separable nature: the

contribution of an agent to the Top_ℓ -cost depends also on the other agents' costs. We overcome this issue by working with the separable proxy function introduced by (Chakrabarty and Swamy 2019). For $z \in \mathbb{R}$, define $z^+ := \max\{z, 0\}$.

Claim 2.3. (Chakrabarty and Swamy 2019) *Let $v \in \mathbb{R}_{\geq 0}^n$ and $\rho \in \mathbb{R}_{\geq 0}$. Then, (a) $\text{Top}_\ell(v) \leq \ell \cdot \rho + \sum_{i=1}^n (v_i - \rho)^+$; and (b) if $v_\ell^\downarrow \leq \rho \leq (1 + \varepsilon)v_\ell^\downarrow$, we have $\ell \cdot \rho + \sum_{i=1}^n (v_i - \rho)^+ \leq (1 + \varepsilon) \cdot \text{Top}_\ell(v)$.*

By identifying a suitable value of ρ , we can work with the separable expression $\sum_i (v_i - \rho)^+$, (where v is the cost vector). This translates ℓ -centrum into a k -median problem, albeit in a non-metric setting, which allows us to exploit ideas used for k -median, for tackling the ℓ -centrum problem.

3 Constant-Factor Distortion Mechanisms

We focus on the setting $A = \mathcal{C}$, that is, the set of candidates and voters are the same. Theorem 3.1 first shows that, for $k \geq 3$, no bounded distortion is possible given only ordinal information. This strengthens the $\Omega(n)$ distortion lower bound of (Anshelevich and Zhu 2018) for $k = 2$. We note that this result also follows from (Caragiannis, Shah, and Voudouris 2022) (who consider a different problem); we include a direct proof in the full version of the paper. Given this, we focus on developing $O(1)$ -distortion mechanisms using a limited number of value queries.

Theorem 3.1. *For k -median with $k \geq 3$, there exists an instance (\mathcal{C}, σ) for which any social choice function has unbounded distortion.*

3.1 Black-Box Reduction: $O(\log k \log n)$ Per-Agent Queries

When the metric is given as input, the ℓ -centrum problem admits various $O(1)$ -factor approximation algorithms. It would be ideal if we could somehow leverage this understanding of the cardinal problem. For instance, if we could somehow reduce the ordinal setting to the cardinal setting, then we could utilize approximation algorithms developed in the cardinal setting to obtain low-distortion mechanisms. A trivial such reduction utilizes queries $d(i, a)$ for every $(i, a) \in \mathcal{C} \times A$, but the question is: can we achieve this end using *substantially* fewer queries. We show that this is indeed possible. We give such a black-box reduction that makes only $O(\log k \log n)$ per-agent queries, while losing only an $O(1)$ -factor in the solution quality; using any $O(1)$ -approximation algorithm for cardinal ℓ -centrum then yields $O(1)$ distortion.

We describe the idea for k -median, i.e., $\ell = n$, which extends with a very minor change to the Top_ℓ setting. We consider a slightly more general setting, where each $i \in \mathcal{C}$ has an integer weight $w_i \geq 0$ denoting the number of agents co-located with i ; so $\sum_{i \in \mathcal{C}} w_i = n$ and the cost of a solution S is $\sum_i w_i d(i, S)$. (This will enable us to handle sparsification seamlessly.) As discussed earlier, we approximate the true underlying metric d^* by a close-enough metric \tilde{d} ; see Mechanism 1. Let $\text{OPT} = \text{OPT}_n(d^*)$ be the optimal value for metric d^* , and $B \in [\text{OPT}, \alpha \text{OPT}]$ be an estimate. For each $i \in \mathcal{C}$ with $w_i > 0$, we consider distance thresholds,

roughly in the range $[\frac{\varepsilon B}{\alpha w_i n}, \frac{B}{w_i}]$, and of the form $\frac{B_{i,0}}{(1+\varepsilon)^r}$ for integer $r \geq 0$, where $B_{i,0}$ is roughly $\frac{B}{w_i}$. ζ , we use binary search on i 's preference profile to find all points with $j \in \mathcal{C}$ with $d(i, j) \leq \tau$. This takes $O(\log n)$ queries per threshold, and hence $O(\log^2 n)$ queries to do this for all ζ 's. Now, replacing $d^*(i, j) \in (\zeta, (1 + \varepsilon)\zeta]$ by any value $\tilde{d}(i, j)$ in this interval incurs only a $(1 + \varepsilon)$ -factor loss; similarly, if $d^*(i, j) \leq \frac{\varepsilon B}{\alpha w_i n}$, then taking $\tilde{d}(i, j) \leq \frac{\varepsilon B}{\alpha w_i n}$ incurs an additive error of at most $w_i \tilde{d}(i, j) \leq \varepsilon \text{OPT}$. So for any \tilde{d} that is consistent with d^* in this fashion, the cost of any solution under \tilde{d} and d^* is roughly the same. We can solve a linear program (LP) to find such a consistent \tilde{d} , and solve k -median with the metric \tilde{d} .

For the Top_ℓ objective, the only change to the above is that we replace w_i by $w'_i = \min\{w_i, \ell\}$; see Remark 3.2.

Mechanism 1: A blackbox reduction

Input: (\mathcal{C}, σ) ; integer weights $\{w_i \geq 0\}_{i \in \mathcal{C}}$ adding up to n ; estimate $B \in [\text{OPT}, \alpha \cdot \text{OPT}]$; ρ -approximation algorithm \mathcal{A} for k -median

- 1: **for** $i \in \mathcal{C}$ with $w_i > 0$ **do**
- 2: Let $B_{i,0} = \rho(1 + 3\varepsilon) \cdot \frac{B}{w_i}$, $q_i = \lceil \log_{1+\varepsilon}(\frac{\alpha w_i B_{i,0} n}{\varepsilon B}) \rceil$
- 3: For each $r = 0, \dots, q_i$, use binary search to compute $S_{i,r} = \{j \in \mathcal{C} : d^*(i, j) \leq B_{i,0}(1 + \varepsilon)^{-r}\}$ in $O(\log n)$ queries
- 4: **end for**
- 5: Solve an LP to find a metric \tilde{d} such that:
 - (1) $\tilde{d}(i, j) \geq B_{i,0}$ for all $i \in \mathcal{C}$, $j \notin S_{i,0}$.
 - (2) $(1 + \varepsilon)^{-(r+1)} B_{i,0} \leq \tilde{d}(i, j) \leq (1 + \varepsilon)^{-r} B_{i,0}$ for all $i \in \mathcal{C}$, $r \in \{0, \dots, q_i - 1\}$, $j \in S_{i,r} \setminus S_{i,r+1}$
 - (3) $\tilde{d}(i, j) \leq \frac{\varepsilon B}{\alpha n \cdot w_i}$ for all $j \in S_{i,q_i}$
- 6: **return** $\mathcal{A}(\mathcal{C}, w, \tilde{d})$

Remark 3.2 (Top_ℓ -objective). The only change for the Top_ℓ objective is that we replace w_i by $w'_i = \min\{w_i, \ell\}$ above. We call the resulting mechanism Mechanism 1- Top_ℓ .

Theorem 3.3. *Let d^* be the true underlying metric.*

- (a) *The center-set F output by Mechanism 1 satisfies $\sum_{j \in \mathcal{C}} w_j d^*(j, F) \leq (\rho(1 + 2\varepsilon + \varepsilon) \text{OPT}_n(d^*))$*
- (b) *Similarly, the output F of Mechanism 1- Top_ℓ satisfies $\text{Top}_\ell(d^*(\mathcal{C}, F|w)) \leq (\rho(1 + 2\varepsilon) + \varepsilon) \text{OPT}_\ell(d^*)$, where $d^*(\mathcal{C}, F|w)$ is the vector in $\mathbb{R}_{\geq 0}^n$ obtained by creating w_i coordinates of value $d(i, F)$ for each $i \in \mathcal{C}$.*

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

Recall that $\text{OPT}_\ell(d^*)$ is the optimal value of ℓ -centrum for the metric d^* . We obtain the estimate B required by these mechanisms using the Boruvka mechanism, which yields $\alpha = n^2$ (see Theorem 1.1). So the combined mechanism, with an $O(1)$ -approximation algorithm for ℓ -centrum, has $O(1)$ distortion and $O(\log^2 n)$ per-agent query complexity.

Improving query complexity. One of the $\log n$ -factors in the $O(\log^2 n)$ per-agent query complexity above comes because we need to do binary search over n agents to compute $S_{i,r}$. To improve this, we sparsify our instance before applying the black-box reduction. We do so by computing a (β, γ) -bicriteria solution for ℓ -centrum (using few queries per agent), where we open at most βk centers and incur cost at most γ times the optimum, and “moving” each agent to its nearest center in the bicriteria solution. Suppose we obtain $\beta, \gamma = O(1)$. Then, we obtain a weighted instance with $O(k)$ points, and we argue that the move to the weighted instance incurs only an $O(1)$ -factor loss. Combining this with the earlier black-box reduction now yields $O(\log k \log n)$ per-agent query complexity.

We compute an $(O(1), O(1))$ -bicriteria solution by extending the algorithm of (Meyerson 2001) for facility location to the ℓ -centrum setting. In *facility location* (FL), any number of facilities may be opened, but every facility has an opening cost f , and we seek to minimize the sum of the assignment costs and the facility-opening costs. Meyerson’s algorithm for FL considers agents appearing online; when the i th client arrives at location x_i , it opens a facility at x_i with probability δ_i/f , where δ_i is the distance from x_i to the closest currently open facility. Meyerson proves, among other things, that when agents appear in a uniform random sequence, for every cluster O^* in an optimal solution with corresponding center $c^* \in \mathcal{C}$, the random solution S returned satisfies $\mathbb{E}[|S \cap O^*|f + \sum_{j \in O^*} d(j, S)] \leq 5f + 8 \sum_{j \in O^*} d(j, o)$. Furthermore, this algorithm yields an $(O(1), O(1))$ -bicriteria solution for k -median if $f = B/k$, where B is a $\Theta(1)$ -estimate of optimal k -median cost.

We adapt Meyerson’s algorithm and analysis to the Top_ℓ -setting, using the separable proxy function $\sum_{j \in \mathcal{C}} (d(j, S) - t)^+$ suggested by Claim 2.3; see Algorithm 1. Viewing $(d(j, S) - t)^+$ as the proxy-cost of agent j , k -clustering to minimize the proxy function gives another type of k -median problem. However, the proxy costs do not satisfy the triangle inequality, and to circumvent complications arising from this, we actually work with the quantity $\delta_j := (d(j, S) - 3t)^+$, and as in Meyerson’s algorithm, open a center at j with probability δ_j/f .

Algorithm 1: Extension of Meyerson’s for ℓ -centrum

Input: Sequence of agents x_1, \dots, x_n , estimate $B \geq OPT$

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

Remark 3.4. We have assumed above that the metric d is given. But if we are only given a preference profile, we can compute δ_i using one value query to i , so the resulting mechanism has *unit* per-agent query complexity.

Theorem 3.5. *If the order of agents is random, the expected*

number of facilities opened by Algorithm 1 is at most $26k$, and the expected cost is at most $15B + 14OPT$.

Thus, given an $O(1)$ -estimate of OPT , Algorithm 1 yields an $(O(1), O(1))$ -bicriteria solution. We do not have such an estimate, but we do have $B' \in [OPT, n^2 \cdot OPT]$, and if we try all powers of 2 in the range $[B'/n^2, B']$, we will find some value in the range $[OPT, 2 \cdot OPT]$. Also, Algorithm 1 may fail with some probability, so we boost its success probability by repetition. Assuming we find the desired bicriteria solution, we move to the weighted instance described earlier, and run the black-box reduction on this weighted instance. Mechanism 2 puts together all of these ingredients; Theorem 3.6 states its performance guarantee.

Mechanism 2: $O(\log k \log n)$ - per-agent query complexity

Input: Preference profile σ , ρ -approximation algorithm \mathcal{A} for ℓ -centrum.

- 1: $S \leftarrow \{S_0\}$ where S_0 is an arbitrary set of k centers
 - 2: B' : Output of Boruvka’s mechanism
 - 3: x_1, \dots, x_n : Randomly shuffled sequence of agents
 - 4: **for** $i = 0, \dots, \lceil \log_2 n^2 \rceil$ **do**
 - 5: $B_i \leftarrow 2^i \cdot B'/n^2$, $f \leftarrow B_i/k$
 - 6: **repeat** $\log(1/\delta)$ **times**
 - 7: S : output of Algorithm 1 with $B = B_i$.
 - 8: **if** $|S| \leq 104k$ **then**
 - 9: $S \leftarrow S \cup \{S\}$; compute $d^*(\mathcal{C}, S)$ using one query per agent
 - 10: **end if**
 - 11: **end**
 - 12: **end for**
 - 13: Let $\bar{S} \leftarrow \arg \min_{S \in \mathcal{S}} \text{Top}_\ell(d^*(\mathcal{C}, S))$. For $i \in \bar{S}$, set $w_i = |\{j \in \mathcal{C} : \text{top}_{\bar{S}}(j) = i\}|$; for all $i \notin \bar{S}$, set $w_i = 0$.
 - 14: **return** Mechanism 1- $\text{Top}_\ell(\bar{S}, \sigma, \{w_j\}_{j \in \mathcal{S}}, B', \mathcal{A})$
-

Theorem 3.6. *Mechanism 2 has $O((\log(1/\delta) + \log k) \log n)$ per-agent query complexity, and achieves $O(1)$ -distortion for the ℓ -centrum problem with probability at least $1 - \delta$.*

3.2 Adaptive Sampling: Per-Agent Query Bounds Independent of n , and Total Query-Complexity

We now develop mechanisms with per-agent query complexity *independent* of n ; moreover, their total query complexity depends on $\text{polylog}(n)$, which is vastly better than the linear dependence on n that follows from Mechanism 2.

The core algorithmic idea here is *adaptive sampling* (Aggarwal, Deshpande, and Kannan 2009; Arthur and Vassilvitskii 2007), which is the following natural idea: we successively choose centers, choosing the next center to add to the current center-set S by sampling a point $i \in \mathcal{C}$ with probability proportional to $d(i, S)$. Aggarwal et al. showed that if we choose $O(k)$ centers this way, then we obtain an $O(1)$ -approximate k -median solution with high probability.

As described, this fails badly for ℓ -centrum, indeed even for k -center. Nevertheless, we show how to extend adaptive sampling in a novel fashion for the ℓ -centrum prob-

lem. Again, the insight is that we can exploit the separable proxy function suggested by Claim 2.3. Intuitively, adaptive sampling works for k -median because, given the current set of centers S , we sample the next point to be added to S with probability proportional to its contribution to the objective, thereby biasing the sampling process towards points that currently incur large cost. The contribution of an agent i to the proxy function given by Claim 2.3 is $(d(i, S) - t)^+$, which suggests that we should sample a point i with probability proportional to this. (Observe that adaptive sampling for k -median corresponds to the special case where $t = 0$.) We show that this does work: for a suitable choice of t , if we choose $O(k)$ centers this way, then we obtain an $O(1)$ -approximate ℓ -centrum solution with high probability, nicely generalizing the guarantee of (standard) adaptive sampling for k -median. In the analysis, we need various new ideas to deal with the fact that distances of the form $(d(i, j) - t)^+$ do not form a metric.

Algorithm 2: Adaptive sampling for ℓ -centrum

Input: instance (\mathcal{C}, d) , parameter $t_\ell \geq 0$

- 1: $S_0 \leftarrow \emptyset$
- 2: **for** $i = 1, \dots, \lfloor 28(k + \sqrt{k}) \rfloor$ **do**
- 3: Sample s_i with probability proportional to $(d(s_i, S_{i-1}) - 2t_\ell)^+$
- 4: Update $S_i \leftarrow S_{i-1} \cup \{s_i\}$.
- 5: **end for**
- 6: **return** $S_{\lfloor 28(k + \sqrt{k}) \rfloor}$

Remark 3.7. We have assumed above that the metric d is given. If we are only given a preference profile, then in each iteration, we make one value query to each agent $j \notin S_{i-1}$ to compute $d(j, \text{top}_{S_{i-1}}(j))$, and thus implement the sampling procedure. The resulting mechanism has $O(k)$ per-agent query complexity.

Let S^* be an optimal solution, and let t_ℓ^* be the ℓ -th largest distance between any voter and their preferred candidate in S^* . When the parameter t_ℓ is sufficiently close to t_ℓ^* , we have the following approximation guarantee for Algorithm 2.

Theorem 3.8. *Let t_ℓ be such that $t_\ell^* \leq t_\ell \leq \max\{(1 + \varepsilon)t_\ell^*, \frac{\varepsilon \cdot OPT}{\ell}\}$, for some $\varepsilon > 0$. Algorithm 2 run with parameter t_ℓ opens at most $56k$ centers, and returns a solution of Top_ℓ -cost at most $35(1 + \varepsilon) \cdot OPT$ with constant probability.*

To compute a suitable t_ℓ (satisfying the conditions of Theorem 3.8), we utilize the estimates B_1 and B_n described in Theorem 1.2 to compute a small set of guesses that contains a suitable choice of t_ℓ . Fix $\varepsilon > 0$ in the sequel.

Claim 3.9. *Let B_1 and B_n be estimates given by Theorem 1.2. Define $\mathcal{T}_1 = \{B_1 \cdot (1 + \varepsilon)^{-r} : r = 0, \dots, \log_{1+\varepsilon}(\frac{2\ell^2}{\varepsilon})\}$ and $\mathcal{T}_2 = \{B_n \cdot (1 + \varepsilon)^{-r} : r = 0, \dots, \log_{1+\varepsilon}(\frac{(8 \ln(k) + 4) \cdot n}{\varepsilon})\}$. There are $t'_\ell \in \mathcal{T}_1$, $t''_\ell \in \mathcal{T}_2$ such that $t_\ell^* \leq t'_\ell, t''_\ell \leq \max\{(1 + \varepsilon)t_\ell^*, \varepsilon \cdot \frac{OPT}{\ell}\}$.*

Combining Algorithm 2 with the set of guesses for t_ℓ^* prescribed by Claim 3.9 yields Mechanism 3, stated below.

Mechanism 3: $\tilde{O}(k \log(\min\{\ell, n/\ell\}))$ per-agent query complexity mechanism

Input: Preference profile σ , ρ -approximation algorithm \mathcal{A} for ℓ -centrum

- 1: $\mathcal{T} \leftarrow \arg \min\{|\mathcal{T}_1|, |\mathcal{T}_2|\}$, $S \leftarrow \emptyset$
 - 2: **for each** $t_\ell \in \mathcal{T}$, repeat $\log(1/\delta)$ times **do**
 - 3: S : output of Algorithm 2 using parameter t_ℓ
 - 4: $\bar{S} \leftarrow S \cup \{S\}$; compute $d^*(\mathcal{C}, S)$ using one query per agent
 - 5: **end for**
 - 6: Let $\bar{S} \leftarrow \arg \min_{S \in \bar{S}} \text{Top}_\ell(d^*(\mathcal{C}, S))$. For $i \in \bar{S}$, set $w_i = |\{j \in \mathcal{C} : \text{top}_{\bar{S}}(j) = i\}|$; for all $i \notin \bar{S}$, set $w_i = 0$.
 - 7: Query $d^*(i, j)$ for all $i, j \in \bar{S}$
 - 8: **return** $\mathcal{A}(\bar{S}, w, d^*)$
-

Theorem 3.10. *The per-agent query complexity of Mechanism 3 is $\tilde{O}(k \log(1/\delta) \log(\min\{\ell, n/\ell\}))$, and it achieves $O(1)$ distortion for ℓ -centrum with probability at least $1 - \delta$.*

Total query complexity. To obtain total query complexity depending on $\text{polylog}(n)$, we change how the adaptive-sampling is implemented in Algorithm 2. Instead of querying agents outside of the current-center set S to obtain $d(\mathcal{C}, S)$, we now construct this vector approximately by querying agents in S . Similar to our black-box reduction, we consider a distance threshold ζ , and find the ring of points $a \in \mathcal{C}$ for which $d(a, S) \in (\zeta, (1 + \varepsilon)\zeta]$. This can be done via binary search on j 's preference profile, for each $j \in S$. We consider geometrically increasing thresholds, using the estimate B_1 obtained from the k -center mechanism to hone in on a $\text{poly}(n)$ -bounded range of distance thresholds. Thus, we need to consider $O(\log n)$ ζ values, and so the total number of queries involved is $O(|S| \log^2 n)$. Since $d(a, S)$ is roughly the same for all points in a ring, we sample by first choosing a ring, and then a uniform point in the ring. With this ring-based implementation of adaptive sampling, we proceed as in Mechanism 3, except that we utilize only B_1 to obtain the candidate set \mathcal{T} of t_ℓ values. The resulting mechanism has total-query-complexity $O(k^2 \log^2 n \log \ell)$, and is described in the full version of this paper.

4 Extension to the Setting $A \neq \mathcal{C}$

Our mechanisms with per-agent query complexity bounds can be extended to the more general setting where $A \neq \mathcal{C}$. While with cardinal information, it is easy enough to reduce this to the earlier case (for instance, by moving agents to the alternatives nearest to them), various challenges arise when we seek to limit the number of value queries because, we cannot query an alternative $a \in A$ for distances to agents.

The changes involved are of two forms. We need to modify how we compute the estimate on OPT using the Boruvka mechanism and the k -center mechanism. The other main change is in adaptive sampling, where when we sample an agent $s \in \mathcal{C}$, we add $\text{top}(s)$ to our center-set. We discuss these changes, and the resulting mechanisms in the full version of the paper.

Acknowledgements

This work was supported in part by C. Swamy’s NSERC Discovery grant.

References

- Abramowitz, B.; Anshelevich, E.; and Zhu, W. 2019. Awareness of Voter Passion Greatly Improves the Distortion of Metric Social Choice. In *Web and Internet Economics - 15th International Conference, WINE 2019*, 3–16.
- Aggarwal, A.; Deshpande, A.; and Kannan, R. 2009. Adaptive Sampling for K-Means Clustering. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 5687, 15–28.
- Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; and Voudouris, A. 2022. Don’t roll the dice, ask twice: the two-query distortion of matching problems and beyond. *Advances in Neural Information Processing Systems*, 35: 30665–30677.
- Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; and Voudouris, A. A. 2020. Peeking behind the Ordinal Curtain: Improving Distortion via Cardinal Queries. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020*, 1782–1789.
- Anari, N.; Charikar, M.; and Ramakrishnan, P. 2023. Distortion in metric matching with ordinal preferences. In *Proceedings of the 24th ACM Conference on Economics and Computation*, 90–110.
- Anshelevich, E.; Bhardwaj, O.; Elkind, E.; Postl, J.; and Skowron, P. 2018. Approximating Optimal Social Choice under Metric Preferences. *Artificial Intelligence*, 264: 27–51.
- Anshelevich, E.; and Postl, J. 2017. Randomized Social Choice Functions Under Metric Preferences. In *J. Artif. Intell. Res.*, volume 58, 797–827.
- Anshelevich, E.; and Zhu, W. 2018. Ordinal Approximation for Social Choice, Matching, and Facility Location Problems Given Candidate Positions. In *Web and Internet Economics - 14th International Conference, WINE 2018*, 3–20.
- Arthur, D.; and Vassilvitskii, S. 2007. K-Means++: The Advantages of Careful Seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, 1027–1035. SIAM.
- Borodin, A.; Halpern, D.; Latifian, M.; and Shah, N. 2022. Distortion in Voting with Top-t Preferences. In *IJCAI*, 116–122.
- Burkhardt, J.; Caragiannis, I.; Fehrs, K.; Russo, M.; Schwegelshohn, C.; and Shyam, S. 2024. Low-Distortion Clustering with Ordinal and Limited Cardinal Information. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 9555–9563.
- Caragiannis, I.; Shah, N.; and Voudouris, A. A. 2022. The metric distortion of multiwinner voting. *Artificial Intelligence*, 313: 103802.
- Chakrabarty, D.; and Swamy, C. 2019. Approximation Algorithms for Minimum Norm and Ordered Optimization Problems. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019*, 126–137. ISBN 978-1-4503-6705-9.
- Charikar, M.; and Ramakrishnan, P. 2022. Metric distortion bounds for randomized social choice. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2986–3004. SIAM.
- Charikar, M.; Ramakrishnan, P.; Wang, K.; and Wu, H. 2024. Breaking the metric voting distortion barrier. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1621–1640. SIAM.
- Chen, X.; Li, M.; and Wang, C. 2020. Favorite-Candidate Voting for Eliminating the Least Popular Candidate in a Metric Space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 1894–1901.
- Faliszewski, P.; Skowron, P.; Slinko, A.; and Talmon, N. 2017. Multiwinner voting: A new challenge for social choice theory. *Trends in computational social choice*, 74(2017): 27–47.
- Gkatzelis, V.; Halpern, D.; and Shah, N. 2020. Resolving the Optimal Metric Distortion Conjecture. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020*, 1427–1438.
- Goel, A.; Hulett, R.; and Krishnaswamy, A. K. 2018. Relating metric distortion and fairness of social choice rules. In *Proceedings of the 13th Workshop on Economics of Networks, Systems and Computation*, 1–1.
- Gonzalez, T. F. 1985. Clustering to Minimize the Maximum Intercluster Distance. *Theoretical computer science*, 38: 293–306.
- Kempe, D. 2020. Communication, Distortion, and Randomness in Metric Voting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 2087–2094.
- Ma, T.; Menon, V.; and Larson, K. 2021. Improving Welfare in One-Sided Matchings Using Simple Threshold Queries. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021*, 321–327.
- Meyerson, A. 2001. Online Facility Location. In *Proc. FOCS’01*, 426–431. ISBN 978-0-7695-1116-0.
- Munagala, K.; and Wang, K. 2019. Improved Metric Distortion for Deterministic Social Choice Rules. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, 245–262.
- Ostrovsky, R.; Rabani, Y.; Schulman, L.; and Swamy, C. 2012. The effectiveness of Lloyd-type methods for the k -means problem. *Journal of the ACM*, 59(6): 28.
- Procaccia, A. D.; and Rosenschein, J. S. 2006. The Distortion of Cardinal Preferences in Voting. In *International Workshop on Cooperative Information Agents*, 317–331. Springer.
- Pulyassary, H. 2021. *Algorithm Design for Ordinal Settings*. Master’s thesis, University of Waterloo.
- Pulyassary, H.; and Swamy, C. 2021. On the Randomized Metric Distortion Conjecture. *arXiv:2111.08698 [cs]*.