The Thirty-Ninth AAAI Conference on Artificial Intelligence (AAAI-25)

Every Bit Helps: Achieving the Optimal Distortion with a Few Queries

Soroush Ebadian and Nisarg Shah

University of Toronto
{soroush,nisarg}@cs.toronto.edu

Abstract

A fundamental task in multi-agent systems is to match n agents to n alternatives (e.g., resources or tasks). This is often done by eliciting agents' ordinal rankings over the alternatives rather than their exact numerical utilities. While this simplifies elicitation, the incomplete information leads to inefficiency, captured by a worst-case measure called *distortion*. Recent work shows that making just a few cardinal utility queries per agent can significantly improve the distortion, with Amanatidis et al. (2024) achieving $O(\sqrt{n})$ distortion with two queries per agent. We generalize their result by achieving $O(n^{1/\lambda})$ distortion with λ queries per agent, for any constant λ, which is optimal up to a constant factor given a previous lower bound by Amanatidis et al. (2022).

We extend this finding to the general social choice problem of selecting one of m alternatives based on n agents' preferences, achieving $O((\min\{n, m\})^{1/\lambda})$ distortion with λ queries per agent, for any constant λ, which is also optimal given prior results. Thus, our work settles open questions regarding the optimal distortion achievable with a fixed number of cardinal value queries in both settings.

1 Introduction

Imagine you are tasked with allocating office spaces in a medical building to a group of doctors. Each doctor provides a ranked list of their preferred offices, and your goal as the manager is to find a matching that maximizes the overall satisfaction, or social welfare, of all doctors. While rankings give you a general sense of their preferences, they lack information about *intensity* of preferences, indicating precisely how much a doctor may value one office over another. If you could ask a few targeted questions to obtain their exact numerical utilities, known as cardinal queries, you may be able to improve the allocation significantly. However, these queries are cognitively burdensome for the doctors to answer, so they must be designed carefully in order to maximize their benefit while minimizing burden on the doctors.

This work addresses the challenge of maximizing social welfare in such settings by leveraging a combination of ordinal rankings and a limited number of cardinal queries. Specifically, we develop algorithms that select a small number of queries to achieve an asymptotically optimal worst-case approximation of the maximum social welfare, a concept known as *distortion* in social choice theory.

The concept of distortion, introduced by Procaccia and Rosenschein (2006), quantifies the loss in social welfare (or other cardinal objectives) due to the lack of exact numerical preferences. Traditionally, distortion has been studied in settings where only ordinal information is available, and the challenge is to approximate the maximum social welfare as closely as possible (see the survey by Anshelevich et al. (2021)). For instance, in one-sided matching problems (Hylland and Zeckhauser 1979), where n agents are matched to n items based on their preferences, the best achievable distortion is known to be $\Theta(\sqrt{n})$, though this requires randomization and normalized values (Filos-Ratsikas, Frederiksen, and Zhang 2014).

Recent work has expanded this framework by considering the trade-offs between the amount of cardinal information gathered and the efficiency of the resulting allocation. Amanatidis et al. (2021, 2022) demonstrated that by allowing mechanisms to ask $\lambda \cdot \log n$ cardinal queries per agent, $O(n^{1/(\lambda+1)})$ distortion can be achieved, even without relying on randomization or normalization. Building on this, Amanatidis et al. (2024) showed that a distortion of $\Theta(\sqrt{n})$ can be achieved by asking only two queries per agent (instead of $\log n$), matching the performance of the best (abovementioned) randomized mechanism that assumes normalized values.

Despite these advancements, our understanding of the trade-offs between using two cardinal queries per agent and a logarithmic number of queries remains limited. This raises an important question:

> *As we go from just two to three or more (but a constant number of) queries per agent, how well can we approximate the optimal social welfare of any one-sided matching?*

Shifting from matching to voting, consider another important decision in the medical building: selecting the location for a new specialized clinic. The doctors must vote on several potential locations, each offering distinct advantages, such as proximity to the emergency room, patient accessibility, or the available space. Each doctor has their own underlying utilities for these locations, influenced by their specialty and the needs of their patients. While rankings provide a general understanding of their preferences, they once again fail to capture preference intensities. By eliciting a few cardinal queries, the decision-maker could potentially improve the selection process, aiming to maximize the collective satisfaction of the group, just as before. However, this voting problem is even less understood than the matching problem. When only two queries per agent are allowed, despite the attempt of Amanatidis et al. (2024), the problem of optimal achievable distortion remains unresolved. This again raises the same question posed earlier, but now for the setting more precisely called *single-winner elections*.

1.1 Our Contributions

We present a novel algorithm for both one-sided matching and single-winner elections, which leverages a limited number of λ cardinal queries per agent to achieve asymptotically optimal distortion bounds when λ is a constant. Tables 1 and 2 provide a summary of our results alongside relevant prior work in one-sided matching and voting, respectively. We impose no restrictions on the utilities, and all our algorithms are deterministic and run in poly-time.[1]

Our work builds on the ideas of Amanatidis et al. (2024) and introduces novel applications of the notion of *stable committees* in multi-winner elections, which is explored by a series of recent works (Aziz et al. 2017; Cheng et al. 2020).

One-sided matching. For the one-sided matching problem, we establish that the optimal distortion with a constant number λ of queries per agent is $O(n^{1/\lambda})$. This significantly generalizes the work of Amanatidis et al. (2024), who only considered two-query algorithms. Our algorithm achieves $O(n^{1/3})$ distortion with just three queries, a result that previously required $O(\log n)$ queries (Amanatidis et al. 2022).

More generally, for any $1 \leqslant \lambda \leqslant \log n$, we achieve distortion of $\lambda \cdot n^{1/\lambda}$ with λ queries. This implies $O(\log n)$ distortion with $\log n$ queries, improving upon the previous best result that required $O(\frac{\log^2 n}{\log \log n})$ queries (Amanatidis et al. 2022). The (asymptotic) optimality of our results for constant λ follows from the $\Omega(n^{1/\lambda})$ lower bound established by Amanatidis et al. (2022).

We also establish the existence of an *exactly* stable committee of matchings by modifying the serial dictatorship algorithm. This result may be of independent interest, as only approximately stable committees exist in committee selection with ranked preferences (Jiang, Munagala, and Wang 2020). Notably, Amanatidis et al. (2024) employ a similar method to satisfy a closely related notion that they introduce.

Single-winner elections. Applying similar techniques, we achieve comparable results for single-winner elections with a distortion bound of $O(\min\{n,m\}^{1/\lambda})$ for constant λ. While Amanatidis et al. (2024) showed $O(\sqrt{m})$ distortion for $m = \Omega(n)$ with two queries, and Amanatidis et al. (2022) achieved $m/2$ distortion, the case of $m = o(n)$, with the number of agents far exceeding the number of candidates, remained open.

[1]All omitted proofs can be found in the full version: https://www.cs.toronto.edu/\%7Enisarg/papers/value-queries.pdf

# Queries	Upper Bounds	Lower Bounds
0 (Ordinal)	-	Unbounded †
1	$O(n)$ †	$\Omega(n)$ †
2	$O(\sqrt{n})$ §	$\Omega(\sqrt{n})$ †
$\lambda = \Theta(1)$	$O(n^{1/\lambda})$	$\Omega(n^{1/\lambda})$ †
$\lambda \leqslant \log n$	$\lambda \cdot n^{1/\lambda}$	$\Omega(n^{1/\lambda}/\lambda)$ †
$\log n$	$O(\log n)$	$\Omega(1)$ †
$z \cdot \log n, z \geqslant 1$	$O(n^{1/(z+1)})$ †	
$O(\log^2 n)$	$O(1)$ †	

Table 1: Summary of our results and prior distortion bounds for one-sided matching with n agents and n alternatives. Gray cells highlight results from Theorem 5. †(Amanatidis et al. 2022), §(Amanatidis et al. 2024).

# Queries	Upper Bounds	Lower Bounds
0 (Ordinal)	-	Unbounded ‡
1	$O(\alpha_{n,m}), O(m)$ ‡	$\Omega(m)$ ‡
2	$O(\sqrt{\alpha_{n,m}})$	$\Omega(\sqrt{m})$ §
$\lambda = \Theta(1)$	$O({\alpha_{n,m}}^{1/\lambda})$	$\Omega(m^{1/\lambda})$ §
$\lambda \leqslant \log(\alpha_{n,m})$	$O(\lambda \cdot {\alpha_{n,m}}^{1/\lambda})$	$\Omega(m^{1/(3\lambda)})$ ⋆
$\log(\alpha_{n,m})$	$O(\log(\alpha_{n,m}))$	$\Omega(1)$
$z \cdot \log m, z \geqslant 1$	$O(m^{1/(z+1)})$ ‡	
$O(\log^2 m)$	$O(1)$ ‡	

Table 2: Summary of our results and prior distortion bounds for single-winner elections with n agents and m candidates. Gray cells (i.e., bounds with $\alpha_{n,m}$) indicate results from Theorem 6, where $\alpha_{n,m} = \min\{n,m\}$. All lower bounds assume $n \geqslant \Omega(m)$. ‡(Amanatidis et al. 2021), §(Amanatidis et al. 2024), ⋆(Caragiannis and Fehrs 2023).

For any $1 \leqslant \lambda \leqslant \log(\min\{n,m\})$, our algorithm achieves distortion $O(\lambda \cdot \min\{n,m\}^{1/\lambda})$. In particular, with $\log(\min\{n,m\})$ queries, we achieve $O(\log(\min\{n,m\}))$ distortion deterministically, comparable to the *randomized* $O(\log m)$ distortion achieved by Caragiannis and Fehrs (2023) using $O(\log m)$ queries.

Notably, our bounds depend on $\min\{n,m\}$ rather than just m as in typical voting distortion results. In Section 4, we discuss the significance of this. For example, it allows modeling the one-sided matching problem as a single-winner election with $n!$ possible matchings as alternatives (i.e., an exponential number of alternatives $m = n!$), yet still achieve an appealing bound of $O(n^{1/\lambda})$ with only a constant factor increase in the distortion bound.

1.2 Related Work

The study of querying beyond ordinal preferences builds on earlier work on distortion using only ordinal information, initiated by Procaccia and Rosenschein (2006) in the context of single-winner elections. Caragiannis and Procaccia (2011); Caragiannis et al. (2017) demonstrate that the optimal distortion achievable by *deterministic* voting rules under normalized valuations is $\Theta(m^2)$. For *randomized* voting rules, Boutilier et al. (2015); Ebadian et al. (2022) identify the best achievable distortion as $\Theta(\sqrt{m})$. Additionally, Caragiannis et al. (2017) explore distortion in multi-winner elections. Borodin et al. (2022) consider scenarios where even less information than ranked votes is used. Distortion is also examined in the *metric voting* framework (Anshelevich and Postl 2017; Anshelevich et al. 2018), with a comprehensive overview provided by Anshelevich et al. (2021).

Some studies relax query-type restrictions, allowing any query that elicits a fixed number of bits from voters, including ordinal elicitation, and focus on optimizing the query format itself (Mandal et al. 2019; Mandal, Shah, and Woodruff 2020) (and Kempe (2020) for metric voting). For single-winner elections, deterministic elicitation requires $\tilde{\Theta}(m/d)$ bits, while randomized elicitation requires $\tilde{\Theta}(m/d^3)$. For committee selection with k candidates, the bounds are $\tilde{\Theta}(m/(kd))$ and $\tilde{\Theta}(m/(kd^3))$, respectively.

Latifian and Voudouris (2024); Ma, Menon, and Larson (2021) investigate threshold approval queries for one-sided matching, while Ebadian, Latifian, and Shah (2023) apply the same elicitation to voting. In the metric voting framework, Ebadian, Halpern, and Micha (2024) propose nearly-optimal algorithms using limited pairwise comparisons per agent, with various constraints on query adaptivity, while Anagnostides, Fotakis, and Patsilinakos (2022) study adaptive algorithms that ask the same set of queries to all agents.

2 Model

Let $[t] := \{1, \dots, t\}$ for $t \in \mathbb{N}$.

One-sided matching. In one-sided matching, there is a set N of n agents and a set A of n alternatives. A matching M is a one-to-one mapping from N to A, pairing each agent with a unique alternative. Each agent $i \in N$ has a valuation function over the alternatives $u_i : A \to \mathbb{R}_{\geqslant 0}$, where $u_i(a)$ denotes the utility agent i receives when matched to alternative a. The utilitarian social welfare of a matching M is denoted by $\mathsf{sw}(M) = \sum_{i \in N} u_i(M(i))$. Similarly, for a subset of agents $N' \subseteq N$, the social welfare of a matching M is $\mathsf{sw}(M \mid N') = \sum_{i \in N'} u_i(M(i))$.

Implicitly utilitarian ordinal matching. For each agent $i \in N$, let $\sigma_i : [n] \to A$ be the preference ranking of agent i induced by the valuation function u_i (ties broken arbitrarily); that is, $u_i(\sigma_i(1)) \geqslant u_i(\sigma_i(2)) \geqslant \dots \geqslant u_i(\sigma_i(n))$. The preference profile $\vec{\sigma} = \{\sigma_i\}_{i \in N}$ is the collection of all agents' preferences. We denote by $\vec{u} \triangleright \vec{\sigma}$ that u_i aligns with σ_i for each agent i. By $a \succ_i a'$ we mean that a appears above a' in i's preference ranking (i.e., i strictly prefers a to a'). We also use $a \succcurlyeq_i a'$ to state either $a = a'$ or $a \succ_i a'$. An *ordinal matching* mechanism returns a matching based only on the preference profile $\vec{\sigma}$.

Value queries. A λ*-query ordinal* matching mechanism, in addition to the preference profile $\vec{\sigma}$, is allowed to ask up to λ *value queries* per agent. Through a value query $Q(i, a)$, the algorithm learns the utility agent i receives from alternative a, i.e., $u_i(a)$.

Distortion. For a utility profile $\vec{u}$, the approximation ratio of a matching M is defined as the ratio of its social welfare to that of the optimal welfare-maximizing matching $\mathtt{OPT}(\vec{u})$, i.e., $\mathsf{sw}(M)/\mathsf{sw}(\mathtt{OPT}(\vec{u}))$. The distortion of a λ-query mechanism $\mathcal{M}$ with respect to a preference profile $\vec{\sigma}$ is its worst-case approximation ratio to the optimal matching over all inputs. That is, for a λ-query matching mechanism $\mathcal{M}$,

$$\mathsf{dist}(\mathcal{M}) = \max_{\vec{u} \triangleright \vec{\sigma}} \frac{\mathsf{sw}\left(\mathcal{M}(\vec{\sigma}, \{Q(i, a_{i,j})\}_{i \in N, j \in [\lambda]})\right)}{\mathsf{sw}(\mathtt{OPT}(\vec{u}))},$$

where $Q(i, a_{i,j})$ is the jth query to i on alternative $a_{i,j} \in A$.

Single-winner voting. In single-winner voting, we have a set N of n agents and a set C of m candidates. We design mechanisms to select one candidate $c \in C$ as the election winner. Similar to above, each agent i has a valuation function $u_i : C \to \mathbb{R}_{\geqslant 0}$, and u_i induces a preference ranking $\sigma_i : [m] \to C$. An *ordinal* voting rule $\mathcal{M}$ receives the preference profile $\vec{\sigma}$ and selects one candidate $\mathcal{M}(\vec{\sigma}) \in C$. We can similarly define a λ-query ordinal voting rule that queries agents about their values for specific candidates and define the distortion of mechanisms.

Local stability in committee selection. In committee selection with ranked preferences, the goal is to select a committee of k candidates that is representative of all agents' preferences. Cheng et al. (2020); Jiang, Munagala, and Wang (2020); Aziz et al. (2017) study a notion of representation called *(local) stability*. A committee is stable if no group of n/k agents (who collectively can select one seat), unanimously prefers some candidate c over all committee members. Since exactly stable committees may not exist, we use an approximation due to Cheng et al. (2020).

To formalize this notion, we present a more general definition using weighted agents. We consider additive weight functions $w : 2^N \to \mathbb{R}_{\geqslant 0}$, i.e., $w(\emptyset) = 0$ and $w(N') = \sum_{i \in N'} w(\{i\})$ for all $N' \subseteq N$. With a slight abuse of notation, we use $w(i) = w(\{i\})$.

Definition 1 (Approximately Stable Committee)**.** *For a committee $X \subset C$ of size k and a candidate $c' \in C \setminus X$, define $V(c', X) = \{i \in N \mid c' \succ_i c, \forall c \in X\}$ to be the set of agents who prefer c' to all in X. Then, X is α-stable if $w(V(c', X)) < \alpha \cdot w(N)/k$ for all $c' \in C \setminus X$.*

For identical weights, the final condition in the definition simplifies to $|V(c', X)| < n/k$ as discussed earlier. Jiang, Munagala, and Wang (2020) establish the following existential result for approximately stable committees.

Theorem 2 (Theorem 1 of Jiang, Munagala, and Wang (2020))**.** *For a ranked preference profile $\vec{\sigma}$, there always exists a $(32+\epsilon)$-stable committee of size $k \in [m]$, which can be computed in time $\mathrm{poly}(n, m, 1/\epsilon)$ for a constant $\epsilon \in (0, 1)$. Furthermore, a $(2 - \epsilon)$-stable committee may not exist for any $\epsilon > 0$.*

For identical weights, Jiang, Munagala, and Wang (2020) show an improved approximation factor of 16 instead of 32.

Algorithm 1 k-Capacity Serial Dictatorship

Input: Preference profile $\vec{\sigma}$, weights $\{w(i)\}_{i\in N}$, and k
Output: A k-capacity mapping

1: $B \leftarrow$ a multiset of k copies of each alternative $a \in A$
2: **for** $i \in N$ in decreasing order of weights w_i **do**
3: Match i to their most preferred alternative $a_i \in B$
4: Remove one copy of a_i from B
5: **end for**
6: **return** $g = \{i \to a_i\}_{i\in N}$

3 Ordinal Matching with Value Queries

In this section, we present a λ-query mechanism with $\lambda \cdot n^{1/\lambda}$ distortion. We first describe a simple algorithm to find a stable "committee" of k matchings in Section 3.1, which is a key subprocedure of the final matching algorithm described in Section 3.2.

3.1 Stable Matching Sets

Amanatidis et al. (2024) define a particular type of assignment called a "sufficiently representative set" that is specific to their two-query mechanism. The crux of this notion and its role in the algorithm, we believe, is closely related to finding a *stable committee* of $\sqrt{n}$ matchings. We extend their algorithm for finding a sufficiently representative set and show a weighted form of k-stable committee of matchings exists and can be computed by a simple algorithm for all $k \in [n]$.

Definition 3 (Stable Set of Matchings). *For an ordinal matching instance with agent weights w, a set of k matchings $X = \{M_1, \dots, M_k\}$ is* stable *if there does not exist another matching M' and agents $N' \subseteq N$ such that $w(N') \geqslant w(N)/k$ and $M'(i) \succ_i M_\ell(i)$ for all $i \in N'$ and $M_\ell \in X$.*

In contrast to Theorem 2, where 2-stable committees may fail to exist, we show a stable set of matchings always exist.

Theorem 4. *For an ordinal matching instance and agents weights w, there always exists a set of k matchings that is stable and can be computed in* $\mathrm{poly}(n)$ *time.*

We achieve Theorem 4 via Algorithm 1 that closely follows the $\sqrt{n}$-Serial Dictatorship algorithm of Amanatidis et al. (2024). The main difference is that we go over the agents in the order of the weights, which enables proving Theorem 4 for arbitrary agent weights and any $k \in [n]$.

k-Capacity serial dictatorship. Algorithm 1 is a serial dictatorship process that starts with a multiset containing k copies of each alternative. Agents then pick their favourite alternative among the remaining ones in an order determined by the non-increasing weights. Agents then make their selections, by picking their favourite among the remaining alternatives, in an order determined by weights in non-increasing order. Since the algorithm begins with k copies, each alternative is mapped to at most k agents. We can decompose such a mapping g into a a set of k matchings where each agent i is mapped to $g(i)$ in at least one of the k matchings.

Lemma 1. *Let $g : N \to A$ be a (k-capacity) mapping where at most k agents are mapped to a single alternative. There exists a set of k matchings $M_1, \dots, M_k$ such that for all $i \in N$, $g(i) = M_\ell(i)$ for some $\ell \in [k]$ and it can be computed in* $\mathrm{poly}(n)$ *time.*

Proof. Let $d_a = \{i \mid g(i) = a\}$ be the number of agents mapped to a. For each alternative a, add the first agent mapped to a (if one exists) to set N_1. Similarly, add the second agent mapped to a (if one existing) to set N_2, and so on for N_3 to N_k. Since the agents in N_1 are all mapped to different alternatives, we can create a matching $M_1 = \{i \to g(i)\}_{i\in N_1}$ and extend it to a complete matching by arbitrarily matching the remaining agents and alternatives. Do the same for N_2 to N_k. This way, we get a set of k matchings where $i \to g(i)$ appears in at least one such matching. □

We are ready to prove Theorem 4.

Proof of Theorem 4. Let $\vec{\sigma}$ be the preference profile of the ordinal matching instance. Take the set of k matching X of Lemma 1 for the output g of Algorithm 1. Suppose by contradiction that X is not stable, and there exists a matching M' and a group of agents $N' \subset N$ with

$$w(N') \geqslant w(N)/k, \tag{1}$$

such that $M'(i) \succ_i M(i)$ for all $M \in X$. Take an agent $i \in N'$. Since i is not matched to $M'(i)$ or alternatives i prefers to $M'(i)$ by Algorithm 1, it must hold that k different agents N_i have appeared before i and were matched to $M'(i)$. Therefore, for all $i' \in N_i$, $w(i') \geqslant w(i)$. This implies

$$w(N_i) \geqslant k \cdot w(i). \tag{2}$$

Since M' is a matching, $M'(i_1) \neq M'(i_2)$ for all $i_1, i_2 \in N'$, and sets $\{N_i\}_{i\in N'}$ are disjoint. Additionally, agent $i_{\min} \in \arg\min_{i\in N'}\{w(i)\}$ that picks last among N', cannot be among any of the N_i's. As otherwise, it implies $i_{\min}$ has appeared before some other agent in N'. Hence,

$$\begin{aligned} w(N) &\geqslant w(i_{\min}) + \sum\nolimits_{i\in N'} w(N_i) \\ &> \sum\nolimits_{i\in N'} w(N_i) \geqslant \sum\nolimits_{i\in N'} k \cdot w(i) = k \cdot w(N'), \end{aligned}$$

where the second inequality follows from $w(i_{\min}) > 0$ and the third from Equation (2). However, this contradicts Equation (1). Hence, X is a stable set of k matchings. Furthermore, the algorithm can be implemented in $O(n^2)$ time. □

3.2 The λ-Query Algorithm

Next, we present our λ-query mechanism that achieves a distortion of $\lambda \cdot n^{1/\lambda}$, which uses Algorithm 1 and its guarantee (Theorem 4) to inform the queries.

The first round of queries. In the first round, since $\lambda = 1$, the algorithm runs a n-capacity serial dictatorship . Since the capacity of n is always unexhausted, the returned map satisfies $g_1(i) = \sigma_i(1)$ and the first query to each agent is about their favourite alternative. We include the formal statement.

Lemma 2. *For each $i \in N$, the first query Algorithm 2 makes to agent i is about their favourite alternative $\sigma_i(1)$, i.e., $g_1(i) = \sigma_1(i)$.*

Algorithm 1 then uses the learnt utilities to decide on the second set of queries.

Algorithm 2 λ-Query Matching Algorithm

Input: Preference profile $\vec{\sigma}$ and k
Output: Matching M

Let $w_1(i) = 1$ for all $i \in N$
for $\ell \in \{1, \dots, \lambda\}$ **do**
  $g_\ell \leftarrow k$-Capacity Serial Dictatorship($\vec{\sigma}$, $\{w_\ell(i)\}_{i \in N}$, $k \leftarrow n^{1-(\ell-1)/\lambda}$)
  Query each agent $i \in N$ of $g_\ell(i)$
  $w_{\ell+1}(i) \leftarrow u_i(g_\ell(i))$, for all $i \in N$
end for
Let $\widetilde{u}_i(a) \leftarrow \max\{u_i(g_\ell(i)) \mid a \succcurlyeq_i g_\ell(i), \ell \in [\lambda]\}$ or 0 if no such queries exists, which is the highest guaranteed utility of i for a learnt from the queries of either a or candidates ranked below a
return social welfare maximizing matching $\widetilde{M}$ based on $\{\widetilde{u}_i\}_{i \in N}$.

Subsequent rounds. For the second query, the algorithm uses the weights $w_2(i) = u_i(g_1(i)) = u_i(\sigma_i(1))$ and computes g_2 returned by a $(n^{1-1/\lambda})$-capacity serial dictatorship with weight vector w_2 and the same preference profile $\vec{\sigma}$. By Lemma 1, we can build a stable set of $n^{1-1/\lambda}$ matchings w.r.t. weights w_2 using g_2. The algorithm makes the second set of queries based on g_2 by asking agent i of their utility for $g_2(i)$.

Similarly, for the third round, the algorithm uses the answers to the second set of queries as the weights for the third round, i.e., $w_3(i) = u_i(g_2(i))$. Finds g_3 by running a $(n^{1-2/\lambda})$-capacity serial dictatorship. Makes the third set of queries based on g_3, uses the utilities as weights for the next round $w_4(i) = u_i(g_3(i))$, computes g_4 by a $(n^{1-3/\lambda})$-capacity serial dictatorship with w_4, and so on.

The final matching. Finally, after making all the λ queries, the algorithm creates a proxy utility profile $\widetilde{u}$ based on the queried utilities. The algorithm sets $\widetilde{u}_i(a)$ to the maximum guaranteed utility learnt by either directly asking $u_i(a)$ or an alternative $a \succ_i a'$ ranked below a. If no information is available, $\widetilde{u}_i(a)$ is set to zero. The algorithm returns a matching with maximum social welfare w.r.t. the utility profile $\widetilde{u}$.

The Analysis. Define $\widetilde{\mathsf{sw}}(M \mid N') = \sum_{i \in N'} \widetilde{u}_i(M(i))$ for all $N' \subseteq N$, to be the social welfare function w.r.t. $\widetilde{u}$. By the way of construction, $\widetilde{u}$ is an underestimation of u, which implies the following lemma.

Lemma 3. *For every matching M, $\mathsf{sw}(M \mid N') \geqslant \widetilde{\mathsf{sw}}(M \mid N')$ for all $N' \subseteq N$.*

Next, we prove a helpful lemma that gives an instance dependent lower bound on the social welfare achieved by the returned matching $\widetilde{M}$.

Lemma 4 (Minimum Welfare Guarantee)**.** *For an ordinal matching instance, the matching $\widetilde{M}$ returned by the λ-query mechanism in Algorithm 2 achieves*

$$\widetilde{\mathsf{sw}}(\widetilde{M}) \geqslant \frac{1}{n^{1-(\ell-1)/\lambda}} \cdot \sum_{i \in N} u_i(g_\ell(i)), \quad \forall \ell \in [\lambda].$$

Proof. Fix an $\ell \in [\lambda]$. Let $\alpha_\ell = n^{1-(\ell-1)/\lambda}$. The mapping g_ℓ is the output of a α_ℓ-capacity serial dictatorship with weights w_ℓ. By Lemma 1, there are α_ℓ matchings $M_1, \dots, M_{\alpha_\ell}$ such that for all agents i, $g_\ell(i) = M_z(i)$ for some $z \in [\alpha_\ell]$. Therefore, we have

$$\sum_{z \in [\alpha_\ell]} \widetilde{\mathsf{sw}}(M_z) \geqslant \sum_{i \in N} u_i(g_\ell(i)).$$

By an averaging argument, we have

$$\max_{z \in [\alpha_\ell]} \widetilde{\mathsf{sw}}(M_z) \geqslant \frac{1}{\alpha_\ell} \sum_{i \in N} u_i(g_\ell(i)).$$

Since, $\widetilde{M}$ is optimal w.r.t. $\widetilde{u}$, we have

$$\widetilde{\mathsf{sw}}(\widetilde{M}) \geqslant \max_{z \in [\alpha_\ell]} \widetilde{\mathsf{sw}}(M_z).$$

From the two inequalities above and by substituting α_ℓ back, we get the sought result. ☐

We now prove the distortion guarantee of Algorithm 2.

Theorem 5. *There is a λ-query ordinal matching mechanism that achieves a distortion of $\lambda \cdot n^{1/\lambda}$ and runs in $\mathrm{poly}(n)$ time.*

Proof. Let $\mathtt{OPT} = \arg\max\{\mathsf{sw}(M) \mid \text{all matchings } M\}$ be the optimal welfare maximizing matching.

At a high-level, we partition the agents into λ groups $N_1, \dots, N_\lambda$ in a particular way and show that $\mathsf{sw}(\mathtt{OPT} \mid N_j) \leqslant n^{1/\lambda} \cdot \mathsf{sw}(\widetilde{M})$ for all $j \in [\lambda]$. Since $\mathsf{sw}(\mathtt{OPT}) = \sum_{j \in [\lambda]} \mathsf{sw}(\mathtt{OPT} \mid N_j)$, we have $\mathsf{sw}(\mathtt{OPT})/\mathsf{sw}(\widetilde{M}) \leqslant \lambda \cdot n^{1/\lambda}$ on any instance, which proves the sought distortion bound.

Partitioning by deviation. From Lemma 2, recall that $g_1(i) = \sigma_i(1)$ for all $i \in N$. No agent strictly prefers $\mathtt{OPT}$ to g_1. Now, let $N_1 = \{i \in N \mid \mathtt{OPT}(i) \succ_i g_2(i)\}$ be the agents who "deviate" from g_2 to $\mathtt{OPT}$, i.e., they (strictly) prefer $\mathtt{OPT}$ to g_2 but not to g_1. Next, let $N_2 = \{i \in N \setminus N_1 \mid \mathtt{OPT}(i) \succ_i g_3(i)\}$ be the agents who prefer $\mathtt{OPT}$ to g_3 but not to g_1 and g_2. Similarly, define

$$N_\ell = \{i \in N \setminus (N_1 \cup \ldots \cup N_\ell) \mid \mathtt{OPT}(i) \succ_i g_{\ell+1}(i)\}$$

for all $\ell \in [\lambda - 1]$ to be the set of agents who deviate from $g_{\ell+1}$ but not g_1 through g_ℓ. Finally, let $N_\lambda = N \setminus (\bigcup_{\ell \in [\lambda-1]} N_\ell)$ be the remaining agents.

Bounding the welfare of N_1. Agents in N_1 prefer $\mathtt{OPT}$ to g_2. By Theorem 4 any such group, including N_1, has a bounded total weight w.r.t. w_2 of

$$w_2(N_1) \leqslant w_2(N)/n^{1-1/\lambda}. \tag{3}$$

Since $w_2(i) = u_i(g_1(i))$ and $g_1(i) \succcurlyeq_i \mathtt{OPT}(i)$ for all $i \in N$,

$$\mathsf{sw}(\mathtt{OPT} \mid N_1) \leqslant w_2(N_1) \text{ and } w_2(N) \leqslant \sum_{i \in N} u_i(g_1(i)).$$

Combined with Eq. (3),

$$\mathsf{sw}(\mathtt{OPT} \mid N_1) \leqslant \frac{1}{n^{1-1/\lambda}} \sum_{i \in N} u_i(g_1(i)).$$

Furthermore, from Lemmas 3 and 4, we have

$$\mathsf{sw}(\widetilde{M}) \geqslant \widetilde{\mathsf{sw}}(\widetilde{M}) \geqslant \frac{1}{n} \sum_{i \in N} u_i(g_1(i)).$$

Therefore, $\mathsf{sw}(\mathtt{OPT} \mid N_1) \leqslant n^{1/\lambda} \cdot \mathsf{sw}(\widetilde{M})$.

Bounding the welfare of N_ℓ. We apply a similar reasoning to all other groups. Fix an $\ell \in [2, \lambda - 1]$. For each agent $i \in N_\ell$, we have $g_\ell(i) \succcurlyeq_i \texttt{OPT}(i) \succ_i g_{\ell+1}(i)$. By Theorem 4, N_ℓ has a bounded total weight w.r.t. $w_{\ell+1}$ of

$$\mathsf{sw}(\texttt{OPT}|N_\ell) \leqslant w_{\ell+1}(N_\ell) \leqslant \frac{w_{\ell+1}(N)}{n^{1-\ell/\lambda}} = \frac{\sum_{i\in N} u_i(g_\ell(i))}{n^{1-\ell/\lambda}}$$

where we used $w_{\ell+1}(i) = u_i(g_\ell(i))$ and $g_\ell(i) \succcurlyeq_i \texttt{OPT}(i)$. From Lemmas 3 and 4, we have

$$\mathsf{sw}(\widetilde{M}) \geqslant \widetilde{\mathsf{sw}}(\widetilde{M}) \geqslant \frac{\sum_{i\in N} u_i(g_\ell(i))}{n^{1-(\ell-1)/\lambda}}. \tag{4}$$

Therefore, $\mathsf{sw}(\texttt{OPT} \mid N_\ell) \leqslant n^{1/\lambda} \cdot \mathsf{sw}(\widetilde{M})$.

For the last group N_λ, since $g_\lambda(i) \succ_i \texttt{OPT}(i)$, instead of arguing with the weights, we directly have $\mathsf{sw}(\texttt{OPT} \mid N_\lambda) \leqslant \sum_{i\in N} u_i(g_\lambda(i))$. Together with Eq. (4) for $\ell = \lambda$, we again have $\mathsf{sw}(\texttt{OPT} \mid N_\lambda) \leqslant n^{1/\lambda} \cdot \mathsf{sw}(\widetilde{M})$.

Distortion bound. Finally, from the above we have

$$\begin{aligned}\mathsf{sw}(\texttt{OPT}) &= \sum\nolimits_{\ell\in[\lambda]} \mathsf{sw}(\texttt{OPT} \mid N_\ell) \\ &\leqslant \sum\nolimits_{\ell\in[\lambda]} n^{1/\lambda} \cdot \mathsf{sw}(\widetilde{M}) = \lambda n^{1/\lambda} \cdot \mathsf{sw}(\widetilde{M}).\end{aligned}$$

Since this holds for any instance, the distortion of the mechanism is at most $\lambda n^{1/\lambda}$. The proof stands complete. □

Implications. For a constant number of queries, i.e., $\lambda = O(1)$, given the $\Omega(n^{1/\lambda})$ lower bound of Amanatidis et al. (2021), Theorem 5 settles the optimal distortion using λ queries to be $\Theta(n^{1/\lambda})$. For $\lambda = \log n$, since $n^{1/\log n} = O(1)$, the bound above translates to a $O(\log n)$ distortion. It is notable that the distortion bound in Theorem 5 is minimized at $O(\log n)$, and it does not achieve a better distortion, i.e., $o(\log n)$ using more queries. Previously, Amanatidis et al. (2021) required $O(\frac{\log^2 n}{\log\log n})$ many queries to achieve a distortion of $O(\log n)$. To achieve a constant distortion, Caragiannis and Fehrs (2023) show that at least $\Omega(\log n)$ many queries is necessary, while Amanatidis et al. (2021) gives an upper bound showing that $O(\log^2 n)$ many queries is enough to achieve $O(1)$ distortion. Settling the gap and finding the optimal number of queries to achieve a constant distortion is exciting open problem for future work.

4 Single Winner Elections

In this section, we turn to the general social choice setting where the goal is to select one out of the m candidates. Existing results for this setting are weaker than their counterparts for one-sided matching. The optimal distortion bound is open even for two queries. The two-query mechanism of Amanatidis et al. (2024) achieves $O(\sqrt{m})$ distortion when $m = \Omega(n)$, however they leave open the setting where $m = o(n)$, i.e., when the number of agents is significantly higher than the number of candidates — which is a prevalent scenario in social decision making. We settle the question for two queries and for a constant number of queries. Our result is a counterpart of Theorem 5 with a similar distortion guarantee for the setting of single-winner elections.

Theorem 6. *There is a λ-query single-winner voting rule that achieves a distortion of $O(\lambda\cdot(\min\{n, m\})^{1/\lambda})$ and runs in $\mathrm{poly}(n, m)$ time.*

More precisely, if a β-approximately stable committee exists for a given instance, the distortion guarantee is $\beta \cdot 1/\lambda \cdot (\min\{n, m\})^{1/\lambda}$.

We provide the proof and the algorithm in the full version and discuss the main differences compared to one-sided matching below. Before discussing the algorithm, we mention the implication of Theorem 6 for when λ is a constant.

Constant number of queries. When λ is a constant, Amanatidis et al. (2024) show a lower bound of $\Omega(m^{1/\lambda})$ on the distortion of any λ-query mechanism. Theorem 6 achieves a distortion of $O(m^{1/\lambda})$ and settles the question of finding the optimal distortion bound in single-winner voting given λ queries (up to constant).

Algorithm differences with the matching setting. The algorithm closely follows Algorithm 2. The main difference is that, instead of invoking Algorithm 1, the voting rule finds a β-approximately stable committee C_ℓ of size $(\min\{n, m\})^{1-(\ell-1)/\lambda}$ in round ℓ. Recall that Theorem 2 due to (Jiang, Munagala, and Wang 2020) shows the existence of such committees for $\beta = 32+\epsilon$ and the algorithm to achieve thereof running in time $\mathrm{poly}(n, m, 1/\epsilon)$. The other slight difference is that, in the ℓth round, agents are asked of their utility for their favourite candidate among C_ℓ. The weights are set similar to Algorithm 2, that is, if agent i is queried of $g_\ell(i)$ in round ℓ, the weights for the next round are set as $w_{\ell+1}(i) = u_i(g_\ell(i))$.

Bits of the analysis. Next, we discuss why we are able to prove a bound that is a function of $\min\{n, m\}$ instead of only the number of candidates m, which is the case for almost all the papers in the distortion literature for voting. For a start, take the case $\lambda = 1$. Suppose we query each agent of their most preferred candidate and learn $u_i(\sigma_i(1))$ as does the algorithm. Based only on the learnt utilities for the top candidates, let c^{Alg} be the candidate with the highest welfare. By a simple averaging argument,

$$\mathsf{sw}(c^{\mathrm{Alg}}) \geqslant \frac{1}{m} \sum\nolimits_{i\in N} u_i(\sigma_i(1))$$

However, if $n < m$, at most n candidates appear at the top of the ranking. By making the averaging argument for those candidates, we can guarantee

$$\mathsf{sw}(c^{\mathrm{Alg}}) \geqslant \frac{1}{n} \sum\nolimits_{i\in N} u_i(\sigma_i(1)).$$

For the optimal candidate $\texttt{OPT}$, we have

$$\mathsf{sw}(\texttt{OPT}) \leqslant \sum\nolimits_{i\in n} u_i(\sigma_i(1)).$$

Therefore, with all the inequalities together, c^{Alg} achieves a distortion of at most

$$\mathsf{sw}(\texttt{OPT})/\mathsf{sw}(c^{\mathrm{Alg}}) \leqslant \min\{n, m\}.$$

This simple observation is key to the algorithm for $\lambda > 1$ and its analysis, specifically we use this in selecting the sizes of the stable committees computed in the subsequent rounds. Intuitively speaking, for the first round, a committee of size $\min\{n, m\}$ is enough to capture the top of the preference profile. Following a geometric progression from $\min\{n, m\}$ to 1, for the subsequent rounds, the sizes of the approximately stable committees is set to $(\min\{n, m\})^{1-1/\lambda}, \ldots, (\min\{n, m\})^{1/\lambda}$.

4.1 Implications for Matching and Beyond

We highlight the significance of Theorem 6's dependence on $\min\{n, m\}$, showing how it leads to Theorem 4 via a black-box reduction, with an added constant factor in the distortion guarantee if we forgo a polytime algorithm.

A combinatorial black-box reduction. Given an ordinal matching instance with $\vec{\sigma}$, create a new preference profile of the same set N of n agents but with a candidate set C of all the $n!$ possible matchings. For each agent i, let σ'_i be their respective preference ranking over all the matchings ordered according to the rank of their match in σ_i, ties broken arbitrarily. All the queries about the "candidates", which are entire matchings, can be implemented by asking an agents utility for their match, which is a valid query in the matching setting. By invoking Theorem 6, since $\min\{n, m\} = \min\{n, n!\} = n$, we immediately get a distortion guarantee of $O(\lambda \cdot n^{1/\lambda})$.

The additional constant factor β (hidden in the O notation) is due to Theorem 2 which states that an exactly stable committee need not always exist. However, since Theorem 4 proves the existence of an exactly stable committee for the matching setting, by invoking Algorithm 1 as the subprocedure of the voting algorithm of Theorem 6, we rederive Theorem 5.

On query efficient resource allocation. In the resource allocation or the fair division setting, the goal is to divide a set G of goods among a set of n agents in a way that is efficient, i.e., makes good use of the available resources, and/or fair. Maximizing the social welfare or achieving an approximation thereof is a wide-studied efficiency objective. In this setting, the set of "candidates" is all the allocations, which is exponentially large, i.e., $|G|^n$ (fix who receives each good).

For a moment, set aside the intractability of considering rankings over the exponentially large set of allocations. Suppose we can access agents' preference rankings over all possible allocations. Theorem 6 shows that, if we can access the rankings, a few *cardinal* queries per agent is enough to achieve nontrivial approximations of the optimal social welfare objective. We consider this an interesting result in two respects.

First, this intractable allocation by "voting" does not assume any structure over the utilities, which is the case for the majority of papers in the field, and not even the *monotonicity* of valuations.[2] This flexibility allows considering *externalities* where one's utility is not only a function of their bundle but also depends on the allocation of others. Notably, most of the literature is focused on settings with no externalities.

[2] A valuation function v is monotone if $v(G'') \leqslant v(G')$ for all $G'' \subseteq G' \subseteq G$.

Moreover, this method is agnostic to the underlying structure of the task at hand, in contrast to the many algorithms specifically designed.

Second, we want to emphasize that eliciting cardinal information can be significantly costlier than ordinal information. Ordinal queries are also more robust, e.g., it is easier to compare two outcomes rather than assigning specific values that may turn out noisy. At one extreme, we have allocation algorithms that can make any number of cardinal queries to agents — which indeed a polytime algorithm makes polynomially many queries. At the other extreme, the black-box reduction above, makes exponentially many ordinal queries but a few (or a constant) number of cardinal queries. We find designing algorithms that make better tradeoffs between eliciting cardinal and ordinal information an important direction for future work.

4.2 On Randomized Voting with No Queries

Next, inspired by the distortion bound of Theorem 6 that is function of $\min\{n, m\}$ rather than only m, we investigate the extent to which this applies to the standard ordinal setting with no value queries. As mentioned before, without any restriction on the utilities, all ordinal voting rules incur an unbounded distortion. The most commonly studied classes of restricted utilities are that of

- *unit-sum*, where each agent $i \in N$ has a total utility of 1 for all candidates combined, i.e., $\sum_{c \in C} u_i(c) = 1$,
- and *unit-range*, where each agent $i \in N$ and candidate $c \in C$, $u_i(c) \in [0, 1]$ such that $u_i(\sigma_i(1)) = 1$ and $u_i(\sigma_i(m)) = 0$.

Ebadian et al. (2022) propose the *stable lottery rule*, demonstrating that it achieves the optimal $O(\sqrt{m})$ distortion (when $n = \Omega(m)$) for a broader class of valuations encompassing both unit-sum and unit-range utilities. By examining scenarios where $n = o(m)$, we highlight a contrast between these utility classes, by proving an upper bound on the distortion for unit-range utilities and a worse lower bound for any randomized rule under unit-sum utilities. We provide the proof in the full version.

Theorem 7. *The stable lottery rule of Ebadian et al. (2022) achieves a distortion of $O(\min\{\sqrt{n}, \sqrt{m}\})$ for unit-range utilities and $O(\min\{n, \sqrt{m}\})$ for unit-sum utilities.*

Theorem 8. *Every randomized voting rule incurs a distortion of at least $\Omega(\min\{n, \sqrt{m}\})$ under unit-sum utilities.*

5 Discussion

The approaches taken by Mandal et al. (2019), Mandal, Shah, and Woodruff (2020), and Kempe (2020), which do not differentiate between ordinal and cardinal queries, and the research assuming that ordinal elicitation is free while cardinal elicitation is very costly, represent more extreme viewpoints. A more balanced approach would involve assigning a cost to ordinal elicitation, with a higher cost for cardinal elicitation, and then optimizing mechanisms within this framework subject to an elicitation budget.

Another limitation of our algorithm is its strong dependence between rounds. Future research could investigate the trade-offs associated with non-adaptive mechanisms.

Acknowledgements

This research was partially supported by an NSERC Discovery grant.

References

Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; and Voudouris, A. A. 2021. Peeking behind the ordinal curtain: Improving distortion via cardinal queries. *Artificial Intelligence*, 296: 103488.

Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; and Voudouris, A. A. 2022. A few queries go a long way: Information-distortion tradeoffs in matching. *Journal of Artificial Intelligence Research*, 74: 227–261.

Amanatidis, G.; Birmpas, G.; Filos-Ratsikas, A.; and Voudouris, A. A. 2024. Don't Roll the Dice, Ask Twice: The Two-Query Distortion of Matching Problems and Beyond. *SIAM Journal on Discrete Mathematics*, 38(1): 1007–1029.

Anagnostides, I.; Fotakis, D.; and Patsilinakos, P. 2022. Metric-distortion bounds under limited information. *Journal of Artificial Intelligence Research*, 74: 1449–1483.

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.; Filos-Ratsikas, A.; Shah, N.; and Voudouris, A. A. 2021. Distortion in Social Choice Problems: The First 15 Years and Beyond. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence Survey Track*, 4294–4301.

Anshelevich, E.; and Postl, J. 2017. Randomized social choice functions under metric preferences. *Journal of Artificial Intelligence Research*, 58: 797–827.

Aziz, H.; Elkind, E.; Faliszewski, P.; Lackner, M.; and Skowron, P. 2017. The Condorcet principle for multiwinner elections: from shortlisting to proportionality. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 84–90.

Borodin, A.; Halpern, D.; Latifian, M.; and Shah, N. 2022. Distortion in voting with top-t preferences. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, 4.

Boutilier, C.; Caragiannis, I.; Haber, S.; Lu, T.; Procaccia, A. D.; and Sheffet, O. 2015. Optimal social choice functions. *Artificial Intelligence*, 227(C): 190–213.

Caragiannis, I.; and Fehrs, K. 2023. Beyond the worst case: Distortion in impartial culture electorate. *arXiv preprint arXiv:2307.07350*.

Caragiannis, I.; Nath, S.; Procaccia, A. D.; and Shah, N. 2017. Subset selection via implicit utilitarian voting. *Journal of Artificial Intelligence Research*, 58: 123–152.

Caragiannis, I.; and Procaccia, A. D. 2011. Voting almost maximizes social welfare despite limited communication. *Artificial Intelligence*, 175(9-10): 1655–1671.

Cheng, Y.; Jiang, Z.; Munagala, K.; and Wang, K. 2020. Group fairness in committee selection. *ACM Transactions on Economics and Computation (TEAC)*, 8(4): 1–18.

Ebadian, S.; Halpern, D.; and Micha, E. 2024. Metric Distortion with Elicited Pairwise Comparisons. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI)*, 2791–2798.

Ebadian, S.; Kahng, A.; Peters, D.; and Shah, N. 2022. Optimized distortion and proportional fairness in voting. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, 563–600.

Ebadian, S.; Latifian, M.; and Shah, N. 2023. The Distortion of Approval Voting with Runoff. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, 1752–1760.

Filos-Ratsikas, A.; Frederiksen, S. K. S.; and Zhang, J. 2014. Social welfare in one-sided matchings: Random priority and beyond. In *Proceedings of the 7th Symposium on Algorithmic Game Theory*, 1–12.

Hylland, A.; and Zeckhauser, R. 1979. The efficient allocation of individuals to positions. *Journal of Political economy*, 87(2): 293–314.

Jiang, Z.; Munagala, K.; and Wang, K. 2020. Approximately stable committee selection. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, 463–472.

Kempe, D. 2020. Communication, distortion, and randomness in metric voting. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 2087–2094.

Latifian, M.; and Voudouris, A. A. 2024. The Distortion of Threshold Approval Matching. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, 2851–2859. Main Track.

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-21*, 321–327. Main Track.

Mandal, D.; Procaccia, A. D.; Shah, N.; and Woodruff, D. 2019. Efficient and thrifty voting by any means necessary. *Advances in Neural Information Processing Systems*, 32.

Mandal, D.; Shah, N.; and Woodruff, D. P. 2020. Optimal communication-distortion tradeoff in voting. In *Proceedings of the 21st ACM Conference on Economics and Computation*, 795–813.

Procaccia, A. D.; and Rosenschein, J. S. 2006. The distortion of cardinal preferences in voting. In *Cooperative Information Agents X: 10th International Workshop, CIA 2006 Edinburgh, UK, September 11-13, 2006 Proceedings 10*, 317–331. Springer.