

Learning-Augmented Metric Distortion via (p, q) -Veto Core

Ben Berger^a, Michal Feldman^{*b}, Vasilis Gkatzelis^{†c}, and Xizhi Tan^{†c}

^aOffchain Labs, inc.; bberger@offchainlabs.com

^bTel-Aviv University and Microsoft ILDC; mfeldman@cs.tau.ac.il

^cDrexel University; {gkatz,xizhi}@drexel.edu

Abstract

In the metric distortion problem there is a set of candidates C and voters V within the same metric space. The goal is to select a candidate minimizing the social cost, defined as the sum of distances of the selected candidate from all the voters, and the challenge arises from the algorithm receiving only *ordinal* input — each voter’s list of candidates ranked by distance — while the objective function is *cardinal*, determined by the underlying metric. The *distortion* of an algorithm is its worst-case approximation factor with respect to the optimal social cost.

A key concept here is the (p, q) -veto core, with $p \in \Delta(V)$ and $q \in \Delta(C)$ being normalized weight vectors representing voters’ veto power and candidates’ support, respectively. The (p, q) -veto core corresponds to a set of winners from a specific class of deterministic algorithms. Notably, the optimal distortion of 3 is obtained from this class, by selecting veto core candidates using uniform p and q proportional to candidates’ plurality scores. Bounding the distortion of other algorithms from this class is an open problem.

Our contribution is twofold. First, we establish upper bounds on the distortion of candidates from the (p, q) -veto core for arbitrary weight vectors p and q . Second, we revisit the metric distortion problem through the *learning-augmented* framework, which equips the algorithm with a (machine-learned) prediction regarding the optimal candidate. The quality of this prediction is unknown, and the goal is to optimize the algorithm’s performance under accurate predictions (known as *consistency*), while simultaneously providing worst-case guarantees under arbitrarily inaccurate predictions (known as *robustness*). We propose an algorithm that chooses candidates from the (p, q) -veto core, using a prediction-guided q vector and, leveraging our distortion bounds, we prove that this algorithm achieves the optimal robustness-consistency trade-off.

^{*}Partially supported by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation program (grant agreement No. 866132), by an Amazon Research Award, and by the NSF-BSF (grant number 2020788).

[†]Partially supported by NSF CAREER award CCF-2047907 and NSF grant CCF-2210502.

1 Introduction

In the metric distortion problem there is a set of candidates \mathcal{C} and a set of voters \mathcal{V} , all residing in the same metric space. This metric defines a distance $d(v, c)$ between each voter $v \in \mathcal{V}$ and candidate $c \in \mathcal{C}$, and the objective is to choose a candidate c with minimum social cost, defined as the total distance of the chosen candidate from all voters, *i.e.*, $\sum_{v \in \mathcal{V}} d(v, c)$. This captures spatial models of voting from the political science literature, such as the Downsian proximity model [26], as well as the classic problem of facility location [21, 37, 24]. Computing the optimal candidate, $c^* = \arg \min_{c \in \mathcal{C}} \sum_{v \in \mathcal{V}} d(v, c)$, given the pairwise distances, $d(v, c)$, is a computationally easy problem, but the challenge in the metric distortion problem is that the algorithm has access only to *ordinal* information regarding the preferences of each voter, *i.e.*, a ranked list of candidates in non-decreasing order of their distance from the voter. This information limitation restricts the designer’s ability to optimize the social cost and the performance of an algorithm is evaluated based on its *distortion*, *i.e.*, its worst-case approximation factor with respect to the optimal social cost.

The metric distortion problem has been the focus of a series of papers [6, 47, 30, 40, 41, 39, 48] which evaluated the distortion of known algorithms and designed new ones, aiming to achieve better distortion bounds. Anshelevich et al. [6] established that no deterministic algorithm can achieve a distortion better than 3, and Gkatzelis et al. [30] eventually provided a deterministic algorithm that matches this bound (see Section 1.2 for details). Even after achieving this tight bound, subsequent work by Kizilkaya and Kempe [40, 41] and Peters [48] has deepened our understanding of the metric distortion problem, leading to the formulation of simpler algorithms that achieve the same bound.

A key concept resulting from this line of work is the (p, q) -*veto core*, where $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$ are normalized weight vectors representing the voters’ veto power and candidates’ support, respectively. Kizilkaya and Kempe [41] showed that the (p, q) -veto core corresponds to a set of candidates that arise as winners under a natural class of algorithms. Notably, all known algorithms that achieve the optimal distortion of 3 always select (p, q) -veto core candidates, where p is uniform over all voters and each $q(c)$ is proportional to the plurality score of candidate c , *i.e.*, proportional to the number of voters that rank c first. However, our understanding of distortion in the veto core is restricted to this specific pair of weight vectors and an important open problem arises, as highlighted by Kizilkaya and Kempe [41]: “The study of the generalized veto core and its connection to metric distortion is very intriguing. In particular, for which initial candidate weights q do candidates in the core guarantee constant (or otherwise small) distortion?” Our first result makes significant progress toward resolving this open problem by establishing upper bounds on the distortion of candidates in the (p, q) -veto core for arbitrary choices of both p and q . We then exhibit the power of these bounds by revisiting the metric distortion problem through the learning-augmented framework and leveraging our bounds to design optimal learning-augmented algorithms.

Learning-augmented framework. Aiming to overcome the limitations of worst-case analysis, a surge of recent work has focused on a new analysis framework for achieving refined bounds using the guidance of (machine-learned) predictions. In this framework, often termed *algorithms with predictions* or *learning-augmented algorithms*, the algorithm is enhanced with a prediction that it can use as a guide toward improved performance. This prediction can take different forms (depending on the information that the designer may benefit from in the setting at hand) and can be generated using historical data or other relevant information available to the designer. However, the quality of this prediction is unknown and cannot be trusted. This framework goes beyond worst-case analysis through bicriteria optimization: the algorithm is simultaneously evaluated based on its performance when the prediction is accurate (known as its *consistency*) as well as its performance when the prediction can be arbitrarily inaccurate (known as its *robustness*).

One extreme choice is to follow the prediction blindly, which is rewarding when it is accurate (leading to good consistency), but can yield very bad outcomes when it is inaccurate (leading to poor robustness). The other extreme is to disregard the prediction, which provides weak consistency guarantees. In general, each learning-augmented algorithm provides a trade-off between robustness and consistency, and the goal is to identify the Pareto frontier between these two measures.

The learning-augmented framework has been applied quite broadly, e.g., to analyze the competitive ratio of online algorithms, the running time of algorithms and data structures, or the performance of mechanisms in multiagent systems (see [42] for a frequently updated list of papers in this rapidly growing literature). In all of these applications, this framework has been used to mitigate information limitations that pose obstacles for the designer.

Since the main challenge in metric distortion is the information gap that the designer faces, this makes it a prime application domain for this framework. Specifically, apart from the rankings of the voters, it is reasonable to assume that the designer may also have access to historical observations regarding the voters' choices in other matters that may correlate with their preferences in the matter at hand, thus providing hints regarding their preferred outcome in the metric space.

1.1 Our Results

Our first main result provides an upper bound for the distortion of any algorithm choosing candidates from the (p, q) -veto core, for arbitrary weight vectors $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$.

Theorem: Any algorithm choosing a candidate c from the (p, q) -veto core has distortion at most

$$1 + \frac{2 \max_v p(v)}{\min_c \frac{q(c)}{\text{plu}(c)}},$$

where $\text{plu}(c)$ is the plurality score of c (the number of voters that rank c at the top of their list).

Note that all the deterministic algorithms known to achieve the optimal distortion of 3 choose candidates from the $(p^{\text{uni}}, q^{\text{plu}})$ -veto core, where $p^{\text{uni}}(v) = \frac{1}{n}$ for all v and $q^{\text{plu}}(c) = \frac{\text{plu}(c)}{n}$ for all c . For these vectors, our bound recovers the optimal distortion of 3. In general, our upper bound makes significant progress towards answering the question posed by Kizilkaya and Kempe [41] by providing a sufficient condition for achieving constant distortion, i.e., that the ratio of $\max_v p(v)$ over $\min_c \frac{q(c)}{\text{plu}(c)}$ is constant. Another implication of our upper bound is that if we let both p and q be uniform, in which case the (p, q) -veto core reduces to the *proportional veto core* introduced by Moulin [46], the resulting distortion is at most $1 + 2 \frac{m}{n} \max_{c \in \mathcal{C}} \text{plu}(c)$. Intuitively, $\max_v p(v)$ captures the “distance” of p from p^{uni} (for which $\max_v p^{\text{uni}}(v) = \frac{1}{n}$) and $\min_c \frac{q(c)}{\text{plu}(c)}$ captures the “distance” of q from q^{plu} (for which $\min_c q^{\text{plu}}(c)/\text{plu}(c) = \frac{1}{n}$). If we use $\lambda = n \cdot \max_v p(v)$ and $\mu = n \cdot \min_c q(c)/\text{plu}(c)$ as two parameters that quantify these “distances,” our upper bound becomes $1 + \frac{2\lambda}{\mu}$, and we complement it by showing that it is tight with respect to μ and λ .

Our second main result, leverages our distortion upper bounds for the (p, q) -veto core, to provide the first analysis of algorithms for the metric distortion problem using the learning-augmented framework. We first consider learning-augmented algorithms that are enhanced with the strongest possible type of prediction, i.e., a prediction regarding the exact cardinal distances between each voter-candidate pair, and we exhibit an inevitable trade-off between the optimal robustness and consistency pairs that any such algorithm can achieve (see Theorem 4.1). We then propose a family of algorithms augmented only with a prediction $\hat{c} \in \mathcal{C}$ regarding who the optimal candidate is, and we prove that despite the weaker prediction that they are provided with, these algorithms are able to match the optimal robustness-consistency trade-off that even the stronger class of algorithms

can achieve (see Theorem 4.3). These algorithms, termed LA_δ , are parameterized by $\delta \in [0, 1]$ and they always return a candidate from the (p, q) -veto core, where p is uniform and q is guided by the prediction, assigning a higher amount of weight to \hat{c} ; the exact amount also depends on the value of δ chosen by the designer.

Theorem: For any $\delta \in [0, 1]$, LA_δ achieves $\frac{3-\delta}{1+\delta}$ -consistency and $\frac{3+\delta}{1-\delta}$ -robustness. Moreover, this is the optimal trade-off. Namely, no deterministic algorithm that is $\frac{3-\delta}{1+\delta}$ -consistent can be strictly better than $\frac{3+\delta}{1-\delta}$ -robust, even for the line metric and just two candidates.

Figure 1 exhibits this optimal trade-off for values of $\delta \in [0, 1)$. Specifically, if the designer is confident regarding the quality of the prediction and willing to relax the robustness guarantee to $\beta > 3$, then choosing the corresponding value of δ yields a consistency guarantee of $(\beta + 3)/(\beta - 1)$. For example, the designer can achieve a consistency of 2 with a guaranteed robustness of 5.

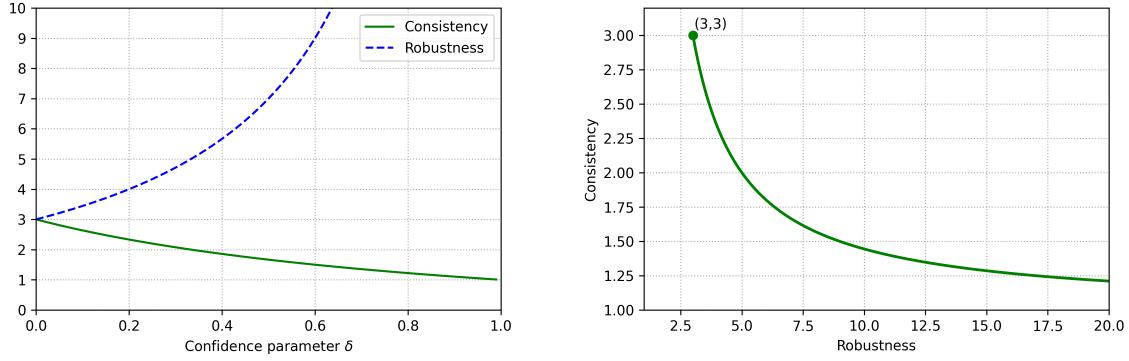


Figure 1: On the left, a plot showing the consistency and robustness bounds achieved by LA_δ as a function of the confidence parameter δ . On the right, a plot of the Pareto-frontier capturing the best possible trade-off between robustness and consistency, all of which can be achieved by LA_δ for the appropriate choice of δ .

A more refined analysis. Consistency and robustness capture two extremes regarding the quality of the prediction, namely, full accuracy and complete arbitrariness, respectively. A more refined analysis provides bounds on the distortion parameterized by the quality of the prediction. In Section 4.3, we provide such refined worst-case bounds on the distortion of LA_δ algorithms. Specifically, we define the prediction error as $\eta := \frac{n \cdot d(\hat{c}, c^*)}{\text{SC}(c^*)/n}$, i.e., the distance between the predicted and the optimal candidates, normalized¹ by the optimal average distance, $\text{SC}(c^*)/n$. We then prove that the distortion of LA_δ with any given δ is at most the bound shown below, which recovers the consistency guarantee of $(3 - \delta)/(1 + \delta)$ for $\eta = 0$, then increases linearly in η up until it reaches the robustness bound of $(3 + \delta)/(1 - \delta)$.

$$\min \left\{ \frac{3 - \delta + 2\delta\eta}{1 + \delta}, \frac{3 + \delta}{1 - \delta} \right\}.$$

Metric spaces with decisive voters. We then refine the distortion bounds further, using the α -decisiveness framework of Anshelevich and Postl [5]. For $\alpha \in [0, 1]$, a voter is α -decisive if her

¹As we discuss in Section 4.3, normalizing the error using the average distance from the optimal candidate is necessary to accurately quantify the quality of the prediction.

distance from her favorite candidate is at most α times her distance from any other candidate. An instance is α -decisive if every voter is α -decisive. We obtain a distortion upper bound of

$$\frac{2 + \alpha - \alpha\delta}{1 + \delta} + \min \left\{ \frac{2\delta\eta}{1 + \delta}, \frac{8\delta}{(1 + \delta)(1 - \delta)} \right\}$$

for α -decisive instances, which recovers our previous bounds when $\alpha = 1$. For $\alpha = 0$, the distance from each voter to her favorite candidate is 0, capturing the well-studied *peer selection* problem, where the set of voters and the set of candidates coincide and each voter ranks herself first.

1.2 Related Work

Deterministic metric distortion. Anshelevich et al. [6] initiated the study of the metric distortion problem and showed that well-known algorithms such as “plurality”, “Borda count”, “ k -approval”, and “veto” admit a distortion that increases linearly with the number of voters or candidates. They also showed that the distortion of the transferable vote algorithm grows logarithmically in the number of candidates, while Copeland’s rule achieves a distortion of 5. They also proved a lower bound of 3 for any deterministic algorithm and conjectured that this bound is tight.

The first deterministic algorithm to guarantee a distortion better than 5 was proposed by Munagala and Wang [47], who achieved a distortion of $2 + \sqrt{5} \approx 4.236$ using a novel algorithm. They also provided a sufficient condition for achieving a distortion of 3. Independently, Kempe [39] derived the same bound of $2 + \sqrt{5}$ using a linear programming duality framework, and provided several alternative formulations of the sufficient condition outlined in [47]. Gkatzelis et al. [30] then introduced the Plurality Matching rule that achieves the optimal distortion of 3.

Even after the tight distortion bound of 3 was achieved, subsequent work provided a better understanding of the problem. Kizilkaya and Kempe [40] achieved the optimal distortion with the Plurality Veto algorithm which is much simpler relative to Plurality Matching, and very recently Kizilkaya and Kempe [41] proposed a refinement of Plurality Veto called Simultaneous Veto that resolves the arbitrary tie-breaking issues in [30] and [40] while maintaining optimal distortion.

Gkatzelis et al. [32] studied the extent to which metric distortion bounds can be combined with distortion bounds for the utilitarian setting — where the voters’ preferences are arbitrary normalized valuations. Abramowitz et al. [1] studied a variant where more information besides the usual preference rankings is available, and Kempe [38] studied the trade-offs between the achievable distortion and the communication complexity of deterministic social choice algorithms.

Metric distortion with randomized algorithms. The tight bound of 3 does not hold when algorithms are allowed to be randomized. Anshelevich and Postl [5] proved that the Random Dictatorship algorithm achieves distortion better than 3, but that tends to 3 as the number of voters grows. Subsequent work by Fain et al. [27] and Kempe [38] identify other algorithms that achieve distortion bounds that tend to 3 (but this time as the number of candidates grows). A lower bound of 2 was also established by [5], which was improved to 2.1126 and 2.06316, independently by Charikar and Ramakrishnan [22], and Pulyassary and Swamy [51] respectively. Charikar et al. [23] also achieved an upper-bound of 2.753, thereby providing the first randomized algorithm with distortion bounded away from 3.

Unlike the deterministic metric distortion problem, the randomized variant remains unresolved even from the traditional worst-case approach. Closing the remaining gap is a major open question in this field and as more advances are achieved in this direction, this provides additional motivation for a refined analysis of that variant through the learning-augmented framework as well.

Learning-augmented framework. The learning-augmented framework has been widely accepted in recent years as a valuable paradigm for the design and analysis of algorithms, aiming to circumvent overly-pessimistic worst-case bounds. In the last five years alone, more than 150 papers have revisited classic algorithmic problems using this framework, with principal examples being online paging [44], scheduling [52], secretary problems [25, 8], optimization problems with covering [12] and knapsack constraints [34], as well as graph problems [9]. We refer the reader to [45] for a survey of some early work and [42] for a more up-to-date list of papers in this area. More closely related to this paper, some recent work has used this framework on social choice problems, where the predictions are regarding a group of agents and their preferences, and the goal is to optimize some social cost or social welfare function. This includes the study of resource allocation problems where agents’ preferences are revealed in an online fashion [13, 14] and settings where the agents are strategic and their preferences may be private [2, 53, 31, 10, 36, 49, 43, 11, 15]. Our work adds to this literature by focusing on settings where access to the agents’ preferences is limited to ordinal information.

Metric spaces with α -decisive voters. The α -decisiveness framework was introduced by Anshelevich and Postl [5] to provide a more refined analysis of distortion bounds, as a function of how strong the agents’ preference is for their top choice, and it has been used in several distortion problems [4, 33, 29]. The special case of $\alpha = 0$ corresponds to the *peer selection* setting, where the set of voters and candidates coincide. This setting has received attention in settings beyond distortion as well [3, 35, 18, 19, 20].

2 Model and Preliminaries

Let (\mathcal{X}, d) be a metric space where $d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_{\geq 0}$ is the distance function that satisfies the following conditions: 1) positive definiteness: $\forall a, b \in \mathcal{X}$, $d(a, b) \geq 0$ and $d(a, b) = 0$ if and only if $a = b$; 2) symmetry: $\forall a, b \in \mathcal{X}$, $d(a, b) = d(b, a)$; and 3) triangle inequality: $\forall a, b, c \in \mathcal{X}$, $d(a, b) + d(b, c) \geq d(a, c)$. Let \mathcal{V}, \mathcal{C} be finite sets of voters and candidates, respectively, that are located in the metric space. We denote by n the number of voters ($n = |\mathcal{V}|$) and by m the number of candidates ($m = |\mathcal{C}|$). For simplicity of notation, we extend d to also operate on the voters and the candidates. That is, we use $d(a, b)$, for $a, b \in \mathcal{V} \cup \mathcal{C}$, to denote the distance between the points in \mathcal{X} where a and b are located. Given a metric d , the *social cost* of a candidate $c \in \mathcal{C}$ is $\text{SC}(c, d) = \sum_{v \in \mathcal{V}} d(v, c)$; we write $\text{SC}(c)$ when the metric distance d is clear from the context.

Preference profile. We refer to triplets $(\mathcal{V}, \mathcal{C}, d)$ as *instances* and the distance between a voter and a candidate quantifies how much the voter prefers that candidate — the closer the better. We say that a voter $v \in \mathcal{V}$ prefers $c \in \mathcal{C}$ over $c' \in \mathcal{C}$, and we write $c \succ_v c'$ if and only if $d(v, c) \leq d(v, c')$. Note that this is a weak preference relation. A given instance $(\mathcal{V}, \mathcal{C}, d)$ induces preference rankings over candidates for each voter. That is, for each $v \in \mathcal{V}$ we have a preference ranking (bijection) $\sigma_v : \mathcal{C} \rightarrow \{1, \dots, |\mathcal{C}|\}$ such that $\sigma_v(c_1) < \sigma_v(c_2)$ implies $c_1 \succ_v c_2$, and we write in this case that d is *aligned* with σ_v , denoted $d \triangleright \sigma_v$. Note that these rankings are not determined uniquely when there are at least two candidates whose distance from v is the same.

A *preference profile* $\sigma := (\sigma_v)_{v \in \mathcal{V}}$ is a tuple of preference rankings for each voter and we say that d is *aligned* with the preference profile σ , denoted as $d \triangleright \sigma$, if $d \triangleright \sigma_v$ for each $v \in \mathcal{V}$. Given σ , we define $\text{top}(v)$ to be the candidate ranked highest by v , and we define $\text{plu}(c)$ to be the number of voters that rank c highest.

Metric distortion. In the metric distortion problem, an algorithm ALG receives as input a preference profile σ which is induced by some $(\mathcal{V}, \mathcal{C}, d)$. Importantly, ALG does not have access to the underlying distance function. The goal is to output a candidate that minimizes the social cost. We denote by $c^*(d) = \arg \min_{c \in \mathcal{C}} \text{SC}(c, d)$ the candidate that minimizes the social cost, *i.e.*, the optimal candidate, breaking ties arbitrarily. The *distortion* of ALG is the worst-case multiplicative approximation it achieves to the optimum, *i.e.*,

$$\text{distortion}(\text{ALG}) = \sup_{\sigma} \sup_{d: d \triangleright \sigma} \frac{\text{SC}(\text{ALG}(\sigma), d)}{\text{SC}(c^*(d), d)}.$$

In addition to using c^* to represent the optimal candidate, we often use a to represent a special candidate with some properties and c to refer to any general candidate.

Generalized veto core Given a preference profile σ , a (p, q) -domination graph for some candidate a is a vertex-weighted bipartite graph with voters on one side and candidates on the other side. It captures how the candidate is evaluated compared with the other candidates. More specifically, an edge (v, c) between some voter v and candidate c exists if voter v prefers candidate a over candidate c . The more edges in the (p, q) -domination graph of some candidate, the more “popular” she is among the voters. We use $\Delta(\mathcal{V})$ and $\Delta(\mathcal{C})$ to denote the probability distribution over \mathcal{V} and \mathcal{C} , respectively.

Definition 2.1 $((p, q)\text{-Domination Graph [30]}).$ *Given a preference profile σ and (normalized) weight vectors $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$, the (p, q) -domination graph of candidate a is the vertex-weighted bipartite graph $G_{p,q}^\sigma(a) = (V, C, E_a, p, q)$, where $(v, c) \in E_a$ if and only if $a \succ_v c$. Vertex $v \in V$ has weight $p(v)$ and vertex $c \in C$ has weight $q(c)$.*

Definition 2.2 $(\text{Fractional perfect Matching}).$ *We say that the (p, q) -domination graph of candidate a , *i.e.*, $G_{p,q}^\sigma(a)$, admits a fractional perfect matching if there exists a weight function $w : E_a \rightarrow \mathbb{R}_{\geq 0}$ such that the total weight of edges incident on each vertex equals the weight of the vertex, *i.e.*, for each $v \in V$ we have $\sum_{c \in C: (v,c) \in E_a} w((v, c)) = p(v)$, and for each $c \in C$ we have $\sum_{v \in V: (v,c) \in E_a} w((v, c)) = q(c)$.*

Kizilkaya and Kempe [41] generalized the notion of proportional Veto Core introduced by Moulin [46] by defining the (p, q) -veto core and they showed that it is equivalent to the set of candidates whose (p, q) -domination graph admits a fractional perfect matching.

Definition 2.3 $((p, q)\text{-Veto Core [41]}).$ *Given a preference profile σ and weight vectors $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$, the (p, q) -Veto Core is the set of candidates a such that $G_{p,q}^\sigma(a)$ admits a fractional perfect matching.*

3 Distortion bounds for (p, q) -Algorithms

In this section, we prove distortion bounds for algorithms that take as input a preference profile σ and return candidates from the (p, q) -veto core of σ . Specifically, in Section 3.1 we establish bounds for arbitrary $q \in \Delta(\mathcal{C})$ paired with a uniform p vector. We then extend this bound in Section 3.2 to arbitrary $p \in \Delta(\mathcal{V})$.

We henceforth refer to the class of algorithms that we will be analyzing as (p, q) -algorithms.

Definition 3.1 $((p, q)\text{-algorithms}).$ *Given arbitrary weight vectors $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$, a (p, q) -algorithm is one that, given a preference profile σ as input, returns a non-empty subset of candidates from the (p, q) -veto core of σ .*

The initial motivation to study such algorithms is due to the work of Munagala and Wang [47], who focused on the special case of weight vectors $(p^{\text{uni}}, q^{\text{plu}})$, where $p^{\text{uni}}(v) = 1/n$ for all $v \in \mathcal{V}$ and $q^{\text{plu}}(c) = \text{plu}(c)/n$ for all $c \in \mathcal{C}$, and showed that if the $(p^{\text{uni}}, q^{\text{plu}})$ -veto core is non-empty, then any $(p^{\text{uni}}, q^{\text{plu}})$ -algorithm achieves an optimal distortion of 3.

Lemma 3.2 ([47, 30]). *If the $(p^{\text{uni}}, q^{\text{plu}})$ -veto core is non-empty, then the distortion of any $(p^{\text{uni}}, q^{\text{plu}})$ -algorithm is at most 3.*

Building upon this lemma, Gkatzelis et al. [30] achieved the optimal distortion bound by proving that the (p, q) -veto core of any preference profile is non-empty for any choice of p and q .

Lemma 3.3 (Ranking-Matching Lemma [30]). *For any preference profile σ and weight vectors $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$, there exists a candidate a whose (p, q) -domination graph $G_{p,q}^\sigma(a)$ admits a fractional perfect matching. Namely, the (p, q) -veto core is non-empty.*

The (p, q) -algorithm proposed by Gkatzelis et al. [30] (PLURALITYMATCHING) identifies an arbitrary candidate in the (p, q) -veto core by just iterating over all candidates and checking whether their (p, q) -domination graph has a fractional perfect matching. Subsequent work by Kizilkaya and Kempe [40, 41] proposed more elegant (p, q) -algorithms that return some subset of candidates in the (p, q) -veto core (e.g., SERIALVETO and SIMULTANEOUSVETO). However, our understanding of the performance of these algorithms with respect to distortion remains limited to the special case of the $(p^{\text{uni}}, q^{\text{plu}})$ weight vectors. Our first result makes significant progress on this matter by establishing a distortion upper bound for (p, q) -algorithms with arbitrary (p, q) weights.

3.1 Distortion Bounds for p^{uni} and General q

In this section, we provide distortion bounds for (p^{uni}, q) -algorithms with arbitrary q . Given any preference profile σ (inducing $\text{plu}(c)$ for each candidate $c \in \mathcal{C}$) and any weight vector $q \in \Delta(\mathcal{C})$, our analysis utilizes the following parameter:

$$\mu(\sigma, q) = \min_{c \in \mathcal{C}} \frac{q(c) \cdot n}{\text{plu}(c)}. \quad (1)$$

We simply write μ when σ and q are clear from the context, and we observe (by an averaging argument) that $\mu \leq 1$. We note that any candidate with $\text{plu}(c) = 0$ can be treated as if $q(c)/\text{plu}(c) = 1$ when computing this parameter value.

Theorem 3.4. Given any $q \in \Delta(\mathcal{C})$, the distortion of any (p^{uni}, q) -algorithm is at most $1 + \frac{2}{\mu}$.

At a high level, for a given preference profile σ , μ quantifies the distance between the given vector q and the vector q^{plu} induced by the instance. More formally, μ represents the fraction of the weight in q that aligns with the plurality score. If this part of weight were the only component present then the distortion for any candidate within the corresponding veto core would be at most 3. Consequently, we decouple the weight into two components and analyze them separately. For the “well-behaved” μ portion of the weight, we utilize the analysis established in prior work. Addressing the “ill-behaved” weight, which comprises the remaining $1 - \mu$ fraction, necessitates a deeper understanding of the (p, q) -domination graph and the implication of fractional perfect matching on the quality of the corresponding candidates.

To prove Theorem 3.4, *i.e.*, the upper bound on $\text{SC}(a, d)/\text{SC}(c^*, d)$ for any agent $a \in (p^{\text{uni}}, q)$ -Veto Core in any metric d , the key technical contribution of this section is establishing an upper bound on the distance between the optimal candidate, c^* , and the corresponding candidate, a , as

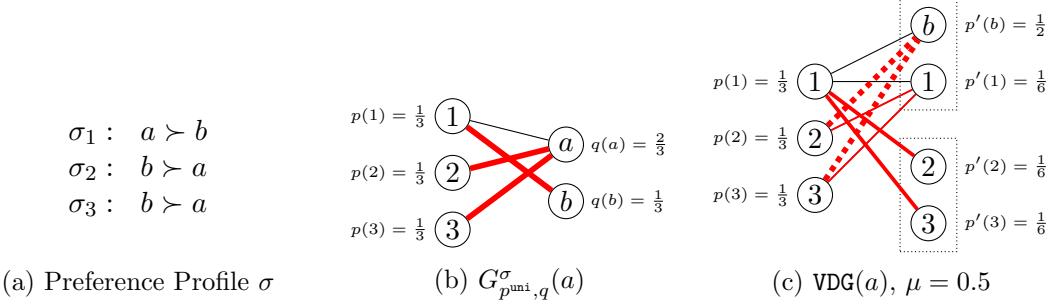


Figure 2: An example of a domination graph and a VDG graph for an instance with preference profile σ . The red edges represent a matching. The value of $\mu(q, \sigma)$ is $\frac{q(b) \cdot 3}{\text{plu}(b)} = \frac{1}{2}$. In the graph $G_{p^{uni},q}^{\sigma}(a)$, the weight of each edge in the matching is $1/3$. In $\text{VDG}(a)$, $w'((2, 1)) = w'((3, 1)) = 1/12$, $w'((1, 2)) = w'((1, 3)) = 1/6$, and $w'((3, b)) = w'((2, b)) = 1/4$. The dotted edges in $\text{VDG}(a)$ represent the “ill-behaved” portion of the weight.

a function of the optimal social cost (see Lemma 3.10). To prove this upper bound, we extend the domination graph of a , $G_{p^{uni},q}^{\sigma}(a)$, to a Voter Domination Graph (VDG graph, see below) for a that captures the conflicts among voter preferences. Leveraging the structure of this graph, we derive a lower bound on the distance between pairs of voters from the optimal candidate, c^* , as stated in Lemma 3.9.

To this end, we first define the *Voter Domination Graph* (VDG) of some candidate a .

Definition 3.5. Let σ be a preference profile, $a \in \mathcal{C}$ be a candidate, and $p \in \Delta(\mathcal{V})$, $q \in \Delta(\mathcal{C})$ be weight vectors such that $G_{p,q}^{\sigma}(a)$ admits a fractional perfect matching. We define the weight vector $p' \in \Delta(\mathcal{V} \cup \{b\})$, where $p'(v) = \frac{\mu}{n}$ and $p'(b) = 1 - \mu$, and the voter’s domination graph (VDG) of a as a bipartite graph $\text{VDG}_{p,p'}^{\sigma}(a) = (V, V \cup \{b\}, E'_a)$, where $(v, v') \in E'_a$ if and only if $a \succ_v \text{top}(v')$, and $(v, b) \in E'_a$ if $(v, c) \in E_a$ and $q(c) > \mu \frac{\text{plu}(c)}{n}$. We simply write $\text{VDG}(a)$ when the parameters are clear from the context.

We note that Definition 3.5 closely resembles Definition 5 from [30]. The key difference lies in the requirement for the graph described in Definition 5 to be integral, with integrality being essential for the analysis to go through. In contrast, our graph may be fractional. In addition, we introduce an extra voter b on the RHS to capture the “ill-behaved” weights which does not exist in [30]’s setting. See Figure 2 for an illustration of VDG. We first observe that for any candidate a whose domination graph $G_{p,q}^{\sigma}(a)$ admits a fractional perfect matching, $\text{VDG}(a)$ also admits a fractional perfect matching.

Lemma 3.6. Consider any candidate a such that $G_{p^{uni},q}^{\sigma}(a)$ admits a fractional perfect matching for some arbitrary q , then $\text{VDG}(a)$ also admits a fractional perfect matching, i.e., there exists a weight function $w' : E'_a \rightarrow \mathbb{R}_{\geq 0}$ such that for each voter $v \in \mathcal{V}$ we have $\sum_{v' \in \mathcal{V} \cup \{b\} : (v, v') \in E'_a} w'((v, v')) = p(v)$ and for each $v' \in \mathcal{V} \cup \{b\}$ we have $\sum_{v \in \mathcal{V} : (v, v') \in E'_a} w'((v, v')) = p'(v')$.

Proof. We prove the statement by constructing $w' : E'_a \rightarrow \mathbb{R}_{\geq 0}$ from w .

$$w'((v, v')) = \frac{\mu}{q(\text{top}(v'))n} \cdot w((v, \text{top}(v'))) \quad (2)$$

$$w'((v, b)) = \sum_{c \in \mathcal{C} : (v, c) \in E_a} \left(1 - \frac{\mu \cdot \text{plu}(c)}{q(c)n}\right) \cdot w((v, c)) \quad (3)$$

First consider any voter v on the LHS, we have:

$$\begin{aligned}
\sum_{v':(v,v') \in E'_a} w'((v, v')) &= \left(\sum_{v' \in \mathcal{V}: (v, v') \in E'_a} w'((v, v')) \right) + w'((v, b)) \\
&= \left(\sum_{c: (v, c) \in E_a} \sum_{v': \text{top}(v') = c} w'((v, v')) \right) + w'((v, b)) \\
&= \sum_{c: (v, c) \in E_a} \frac{\mu \cdot \text{plu}(c)}{q(c)n} \cdot w((v, c)) + \sum_{c: (v, c) \in E_a} \left(1 - \frac{\mu \cdot \text{plu}(c)}{q(c)n} \right) \cdot w((v, c)) \\
&= \sum_{c: (v, c) \in E_a} \left(1 - \frac{\mu \cdot \text{plu}(c)}{q(c)n} + \frac{\mu \cdot \text{plu}(c)}{q(c)n} \right) \cdot w((v, c)) \\
&= \sum_{c: (v, c) \in E_a} w((v, c)) = p(v) \tag{by Def. 2.2}
\end{aligned}$$

Now consider any voter v' on the RHS, and let $c = \text{top}(v')$. We have:

$$\begin{aligned}
\sum_{v: (v, v') \in E'_a} w'((v, v')) &= \sum_{v: (v, c) \in E_a} \frac{\mu}{q(c)n} w((v, c)) \\
&= \frac{\mu}{q(c)n} \sum_{v: (v, c) \in E_a} w((v, c)) \\
&= \frac{\mu}{q(c)n} \cdot q(c) \tag{by Def. 2.2} \\
&= \frac{\mu}{n} = p'(v')
\end{aligned}$$

Finally, we consider the added voter b on the RHS. We have:

$$\begin{aligned}
\sum_{v: (v, b) \in E'_a} w'((v, b)) &= \sum_{v: (v, b) \in E'_a} \sum_{c: (v, c) \in E_a} \left(1 - \frac{\mu \cdot \text{plu}(c)}{q(c)n} \right) \cdot w((v, c)) \\
&= \sum_{c \in \mathcal{C}} \sum_{v: (v, c) \in E_a} \left(1 - \frac{\mu \cdot \text{plu}(c)}{q(c)n} \right) \cdot w((v, c)) \\
&= \sum_{c \in \mathcal{C}} \left(1 - \frac{\mu \cdot \text{plu}(c)}{q(c)n} \right) \cdot q(c) \tag{by Def. 2.2} \\
&= 1 - \mu \sum_{c \in \mathcal{C}} \frac{\text{plu}(c)}{n} = 1 - \mu = q'(b) \quad \square
\end{aligned}$$

The Corollary below follows from Lemma 3.6.

Corollary 3.7. $\sum_{(v, v') \in E'_a: v' \in \mathcal{V}} w'((v, v')) = \mu$.

We first observe the following relations between voters v, v' and candidate a , for every pairs of v, v' with $(v, v') \in E'_a$.

Observation 3.8. Consider any candidate a such that $G_{p,q}^\sigma(a)$ admits a fractional perfect matching for p^{uni} and some arbitrary q . If $(v, v') \in E'_a$ in the VDG of candidate a , $\text{VDG}(a)$, then for any $d \triangleright \sigma$,

$$d(v, a) \leq d(v, c^*) + 2d(v', c^*).$$

Proof. Consider any $v, v' \in \mathcal{V}$ such that $(v, v') \in E'_a$, let $c = \text{top}(v')$. We have

$$\begin{aligned}
d(v, a) &\leq d(v, c^*) + d(c^*, a) && \text{(triangle ineq.)} \\
&\leq d(v, c^*) + d(c^*, c) && \text{(by Def. 2.1)} \\
&\leq d(v, c^*) + d(c^*, v') + d(v', a) && \text{(triangle ineq.)} \\
&\leq d(v, c^*) + d(v', c^*) + d(v', c^*) && \text{(since } c = \text{top}(v')\text{)} \\
&\leq d(v, c^*) + 2d(v', c^*). && \square
\end{aligned}$$

We now make the crucial observation that if candidate a is in the (p, q) -veto core, then for every edge $(v, v') \in E'_a$ (i.e., for every edge that does not involve b in the VDG of candidate a), we can derive a lower bound on the combined contribution of voters v and v' to the social cost of the optimal candidate, c^* . Intuitively, if v prefers a over the candidate closest to v' (this is the meaning of the assumption that $(v, v') \in E'_a$), then v must be far enough from that candidate and hence from v' as well. In turn, if one of v or v' is close to c^* , then the other voter must be far enough from c^* .

Lemma 3.9. *Consider any candidate a such that $G_{p,q}^\sigma(a)$ admits a fractional perfect matching for p^{uni} and some arbitrary q . If $(v, v') \in E'_a$ in the VDG of candidate a , $\text{VDG}(a)$, then for any $d \triangleright \sigma$,*

$$d(v, c^*) + d(v', c^*) \geq \frac{d(c^*, a)}{2}.$$

Proof. Consider any v , by the triangle inequality we have

$$d(c^*, a) \leq d(a, v) + d(v, c^*) \Rightarrow d(a, v) \geq d(c^*, a) - d(v, c^*). \quad (4)$$

Consider any $v' \in \mathcal{V}$ such that $(v, v') \in E'_a$, let $c = \text{top}(v')$. We have

$$d(c^*, c) \geq d(c, v) - d(v, c^*) \geq d(a, v) - d(v, c^*) \geq d(c^*, a) - 2d(v, c^*), \quad (5)$$

where the first inequality holds by the triangle inequality, the second holds since $d(c, v) \geq d(a, v)$ by Definition 3.5, and the last inequality is using (4). Now consider voter v' , we have

$$\begin{aligned}
d(c^*, v') + d(v', c) &\geq d(c^*, c) && \text{(by triangle inequality)} \Rightarrow \\
2d(c^*, v') &\geq d(c^*, c) && \text{(since } c = \text{top}(v') \succ_{v'} c^*\text{)} \Rightarrow \\
2d(c^*, v') &\geq d(c^*, a) - 2d(v, c^*) && \text{(using Inequality (5))} \Rightarrow \\
d(c^*, v) + d(c^*, v') &\geq \frac{d(c^*, a)}{2}. && \square
\end{aligned}$$

We now show the key lemma that connects the social cost of the optimal candidate c^* with the distance between the candidate a who admits a fractional perfect matching and c^* .

Lemma 3.10. *Consider any candidate a in the (p^{uni}, q) -veto core of σ for some arbitrary q . Then, for any metric $d \triangleright \sigma$,*

$$\text{SC}(c^*, d) \geq \left(\frac{\mu}{1 + \mu} n \right) \frac{d(c^*, a)}{2}.$$

Proof. Consider the optimal social cost

$$\text{SC}(c^*, d) = \sum_{v \in \mathcal{V}} d(c^*, v)$$

$$\begin{aligned}
&= \sum_{v \in \mathcal{V}} \frac{1}{p(v) + p'(v)} \cdot (p(v) + p'(v)) \cdot d(c^*, v) \\
&= \frac{n}{1 + \mu} \sum_{v \in \mathcal{V}} (p(v) + p'(v)) \cdot d(c^*, v) \\
&= \frac{n}{1 + \mu} \left(\sum_{v \in \mathcal{V}} p(v) \cdot d(c^*, v) + \sum_{v' \in \mathcal{V}} p'(v') \cdot d(c^*, v') \right) \\
&= \frac{n}{1 + \mu} \left(\sum_{v \in \mathcal{V}} \sum_{v' \in \mathcal{V} \cup \{b\}} w((v, v')) \cdot d(c^*, v) + \sum_{v' \in \mathcal{V}} \sum_{v \in \mathcal{V}} w((v, v')) \cdot d(c^*, v') \right) \quad (\text{by Lem. 3.6}) \\
&\geq \frac{n}{1 + \mu} \left(\sum_{v \in \mathcal{V}} \sum_{v' \in \mathcal{V}} w((v, v')) \cdot d(c^*, v) + \sum_{v' \in \mathcal{V}} \sum_{v \in \mathcal{V}} w((v, v')) \cdot d(c^*, v') \right) \quad (\text{removing } b) \\
&\geq \frac{n}{1 + \mu} \sum_{(v, v') \in E'_a} w'((v, v')) \cdot (d(c^*, v) + d(c^*, v')) \quad (\text{rearranging the sum}) \\
&\geq \frac{n}{1 + \mu} \sum_{(v, v') \in E'_a} w'((v, v')) \cdot \frac{d(c^*, a)}{2} \quad (\text{by Lem. 3.9}) \\
&\geq \frac{n}{1 + \mu} \cdot \mu \cdot \frac{d(c^*, a)}{2} \quad (\text{by Corollary 3.7}) \\
&= \left(\frac{\mu}{1 + \mu} n \right) \frac{d(c^*, a)}{2}. \quad \square
\end{aligned}$$

We are now ready to prove Theorem 3.4.

Proof of Theorem 3.4. Consider any candidate a in the (p^{uni}, q) -veto core of σ for p^{uni} and some q , i.e., a is any candidate that could be returned by a (p^{uni}, q) -algorithm. We show that for any metric $d \triangleright \sigma$,

$$\text{SC}(a, d) \leq \left(1 + \frac{2}{\mu} \right) \cdot \text{SC}(c^*, d),$$

where $\mu = \min_{c \in \mathcal{C}} \frac{q(c) \cdot n}{\text{plu}(c)}$. The social cost of this candidate a can be upper bounded as follows:

$$\begin{aligned}
\text{SC}(a, d) &= \sum_{v \in \mathcal{V}} d(v, a) \\
&= \sum_{v \in \mathcal{V}} n \cdot p(v) \cdot d(v, a) \quad (\text{since } p \text{ is uniform}) \\
&= n \sum_{v \in \mathcal{V}} \left(\sum_{v' \in \mathcal{V}} w'((v, v')) + w'((v, b)) \right) \cdot d(v, a) \\
&= n \sum_{v \in \mathcal{V}} \sum_{v' \in \mathcal{V}} w'((v, v')) \cdot d(v, a) + n \sum_{v \in \mathcal{V}} w'((v, b)) \cdot d(v, a) \\
&= n \sum_{v \in \mathcal{V}} \sum_{v' \in \mathcal{V}} w'((v, v')) \cdot (d(v, c^*) + 2d(v', c^*)) + n \sum_{v \in \mathcal{V}} w'((v, b)) \cdot d(v, a) \quad (\text{by Obs. 3.8}) \\
&\leq n \sum_{v \in \mathcal{V}} \sum_{v' \in \mathcal{V}} w'((v, v')) \cdot (d(v, c^*) + 2d(v', c^*)) + \sum_{v \in \mathcal{V}} w'((v, b)) \cdot (d(v, c^*) + d(c^*, a))
\end{aligned}$$

$$\begin{aligned}
&= n \sum_{v \in \mathcal{V}} \sum_{v' \in \mathcal{V} \cup \{b\}} w'((v, v')) \cdot d(v, c^*) + n \sum_{v' \in \mathcal{V}} \sum_{v \in \mathcal{V}} w'((v, v')) \cdot 2d(v', c^*) + n \sum_{v \in \mathcal{V}} w'((v, b)) \cdot d(c^*, a) \\
&= \sum_{v \in \mathcal{V}} np(v) \cdot d(v, c^*) + n \sum_{v' \in \mathcal{V}} np'(v') \cdot 2d(v', c^*) + n(1 - \mu) \cdot d(c^*, a). \quad (\text{by Lem. 3.6}) \\
&= \sum_{v \in \mathcal{V}} d(v, c^*) + \sum_{v' \in \mathcal{V}} 2\mu \cdot d(v', o) + n(1 - \mu) \cdot d(c^*, a) \quad (\text{by Def. 3.5}) \\
&= (1 + 2\mu) \text{SC}(c^*, d) + n(1 - \mu) 2 \frac{1 + \mu}{\mu} \text{SC}(c^*, d) \\
&= \left(1 + \frac{2}{\mu}\right) \cdot \text{SC}(c^*, d). \quad \square
\end{aligned}$$

3.2 Distortion Bounds for (p, q) -algorithms with General p and q

We now extend the distortion upper bound we obtained in the previous section for p^{uni} to prove a distortion upper bound for general p (and q). Specifically, we reduce any instance with some arbitrary p and q to an instance with p^{uni} and a new q' such that the (p^{uni}, q') -veto core is a superset of the original (p, q) -veto core. To do so, we show that if — for some candidate a and arbitrary p, q — $G_{p,q}^{\sigma}(a)$ admits a fractional perfect matching, then the new q' that we construct has a fractional perfect matching in the domination graph $G_{p^{\text{uni}}, q'}^{\sigma}(a)$, and we carefully bound the change in μ in the transition from q to q' .

The high level idea is to add weights on both sides so as to “re-balance” the given p vector, and re-normalize the weight. We introduce a new parameter λ to capture the amount of extra weight that (relatively) needs to be added to re-balance p . All the proofs missing from this section can be found in Appendix A.

Lemma 3.11. *Consider any candidate a in the (p, q) -veto core of σ for for some arbitrary $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$, then there exists some $q' \in \Delta(\mathcal{C})$ such that $G_{p^{\text{uni}}, q'}^{\sigma}(a)$ also admits a fractional perfect matching. In addition, if we let $\lambda = n \max_v p(v)$, we have*

$$\mu(\sigma, q') \geq \frac{\mu(\sigma, q)}{\lambda}.$$

Combining Theorem 3.4 and Lemma 3.11 the corollary below follows.

Corollary 3.12. *Given a preference profile σ and any $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$, the distortion of any (p, q) -algorithm is at most*

$$1 + \frac{2 \max_v p(v)}{\min_c \frac{q(c)}{p \text{lu}(c)}}$$

3.3 Distortion Lower Bounds

We now show that our upper bound is tight with respect to parameters μ and λ , i.e., for any given $\mu \in (0, 1]$ and $\lambda \in [1, n]$ there exists a metric d and vectors $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$ that yield the given μ and λ , such that some candidate a in the (p, q) -veto core with distortion at least $1 + \frac{2\lambda}{\mu}$.

Theorem 3.13. *Given any $\mu \in (0, 1]$ and $\lambda \in [1, n]$, there exists a preference profile σ and weight vectors $q \in \Delta(\mathcal{C})$ with $\mu = \min_{c \in C} \frac{q(c) \cdot n}{p \text{lu}(c)}$ and $p \in \Delta(\mathcal{V})$ with $\lambda = n \max_v p(v)$, such that for any $\epsilon > 0$ there is a metric $d \triangleright \sigma$ and a candidate a in the (p, q) -veto core of σ with*

$$\text{SC}(a, d) \geq \left(1 + \frac{2\lambda}{\mu} - \epsilon\right) \text{SC}(c^*, d).$$

4 Optimal Distortion with Predictions

We now provide our second main result, which is a tight analysis of the metric distortion problem through the learning-augmented framework. Specifically, we provide an optimal learning-augmented algorithm for metric distortion and we achieve this by leveraging the results of the previous section. While this demonstrates the applicability of our first main result, we believe that our results within the learning-augmented framework hold significant independent interest beyond being merely an application of our first result.

In the learning-augmented framework, the regular input that the algorithm receives is augmented with a prediction \hat{p} that may provide the algorithm with some guidance regarding how to approach this instance. For example, in the metric distortion problem, \hat{p} could be a prediction regarding the metric d , i.e., all the pairwise distances between agents and voters. Crucially, this prediction can be arbitrarily inaccurate, and the goal is to design algorithms guided by this prediction that guarantee better distortion whenever it happens to be accurate (*consistency*), while maintaining some worst-case distortion guarantee even if the prediction is (arbitrarily) inaccurate (*robustness*). These algorithms receive as input both a preference profile σ and prediction \hat{p} , and we use $d \triangleright (\sigma, \hat{p})$ to denote the restriction to metrics d for which the prediction \hat{p} is accurate.

Formally, the consistency of ALG is the distortion it guarantees when \hat{p} is accurate, i.e.,

$$\text{consistency}(\text{ALG}) = \sup_{\sigma} \sup_{d: d \triangleright (\sigma, \hat{p})} \frac{\text{SC}(\text{ALG}(\sigma, \hat{p}), d)}{\text{SC}(c^*(d), d)}.$$

The robustness of ALG is the distortion it guarantees irrespective of the prediction accuracy, i.e.,

$$\text{robustness}(\text{ALG}) = \sup_{\sigma, \hat{p}} \sup_{d: d \triangleright \sigma} \frac{\text{SC}(\text{ALG}(\sigma, \hat{p}), d)}{\text{SC}(c^*(d), d)}.$$

In Section 4.1, we consider algorithms augmented with the strongest type of prediction, i.e., a prediction \hat{p} regarding all the pairwise distance among voters and candidates, and we prove an impossibility result regarding the robustness and consistency guarantees that any such algorithm can achieve. Then, in Section 4.2, we consider algorithms augmented with a much weaker prediction: a prediction regarding who the optimal candidate is. We propose a family of (p, q) -algorithms that use the prediction to determine the q vector, and prove that the robustness-consistency trade-off these algorithms achieve exactly matches the impossibility result shown in Section 4.1. Therefore, despite the fact that they are provided with a weaker type of prediction, these algorithms achieve the optimal distortion and consistency that algorithms with the strongest prediction could achieve. Finally, we provide a more refined analysis on the distortion as a function of the prediction accuracy (Section 4.3), and as a function of the voters' decisiveness (Section 4.4).

4.1 Impossibility Result for Algorithms with Full-Metric Predictions

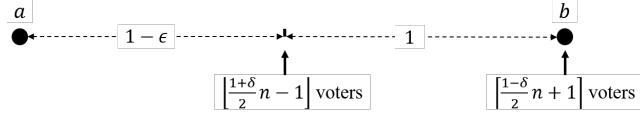
We now show that even if an algorithm is augmented with the strongest type of prediction (a prediction regarding all the pairwise distances), it cannot simultaneously achieve $\frac{3-\delta}{1+\delta}$ -consistency and better than $\frac{3+\delta}{1-\delta}$ robustness for any $\delta \in [0, 1)$.

Theorem 4.1. For any $\delta \in [0, 1)$, let ALG be a deterministic algorithm augmented with a prediction regarding the whole metric that is $\frac{3-\delta}{1+\delta}$ -consistent. Then, for any $\beta < \frac{3+\delta}{1-\delta}$, ALG is not β -robust.

Proof. Let $r(n, \epsilon) := (3 + \delta - \frac{4}{n} - 2\epsilon) / (1 - \delta + \frac{4}{n})$, where δ and β are as in the theorem statement. Note that for every $n, \epsilon > 0$ we have $r(n, \epsilon) < \frac{3+\delta}{1-\delta}$. Since this function is continuous in both n

and ϵ , and $\lim_{n \rightarrow \infty, \epsilon \rightarrow 0^+} r(n, \epsilon) = \frac{3+\delta}{1-\delta}$, then there exist n, ϵ for which $\beta < r(n, \epsilon) < \frac{3+\delta}{1-\delta}$. Fix any n, ϵ that satisfy this, and also satisfy $1/n - \epsilon > 0$ (since $1/n > 0$, we can always choose a small enough value for ϵ that satisfies both inequalities).

Consider the instance $\mathcal{I}_1 = (\mathcal{V}, \mathcal{C}, d)$ consisting of two candidates a, b whose distance from one another is $2 - \epsilon$. The voters are located in the metric space such that $\lceil \frac{1-\delta}{2}n + 1 \rceil$ of them are placed on b , and the rest (of which there are $\lfloor \frac{1+\delta}{2}n - 1 \rfloor$) are placed almost at the halfway point with a slight inclination towards a , such that their distance from b is 1 and their distance from a is $1 - \epsilon$.

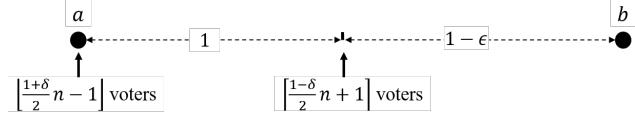


Note that b is the optimal candidate, and a achieves distortion equal to:

$$\begin{aligned} \frac{\text{SC}(a, d)}{\text{SC}(b, d)} &= \frac{\lfloor \frac{1+\delta}{2}n - 1 \rfloor(1 - \epsilon) + \lceil \frac{1-\delta}{2}n + 1 \rceil(2 - \epsilon)}{\lfloor \frac{1+\delta}{2}n - 1 \rfloor \cdot 1} \\ &= \frac{(\lfloor \frac{1+\delta}{2}n - 1 \rfloor + \lceil \frac{1-\delta}{2}n + 1 \rceil)(1 - \epsilon) + \lceil \frac{1-\delta}{2}n + 1 \rceil \cdot 1}{\lfloor \frac{1+\delta}{2}n - 1 \rfloor} \\ &= \frac{n(1 - \epsilon) + \lceil \frac{1-\delta}{2}n + 1 \rceil \cdot 1}{\lfloor \frac{1+\delta}{2}n - 1 \rfloor} \geq \frac{n(1 - \epsilon) + (\frac{1-\delta}{2}n + 1)}{\frac{1+\delta}{2}n} \\ &= \frac{(3 - \delta)\frac{n}{2} + 1 - \epsilon n}{(1 + \delta)\frac{n}{2}} = \frac{3 - \delta + \frac{2}{n} - 2\epsilon}{1 + \delta} > \frac{3 - \delta}{1 + \delta} \end{aligned}$$

Therefore, if **ALG** is provided as input the preference profile σ induced by $(\mathcal{V}, \mathcal{C}, d)$ and the prediction provided to the mechanism is accurate (i.e., it accurately predicts $(\mathcal{V}, \mathcal{C}, d)$), then to achieve a consistency of $\frac{3-\delta}{1+\delta}$, **ALG** must output b .

Now, consider the instance $\mathcal{I}_2 = (\mathcal{V}, \mathcal{C}, d')$, with the same sets of voters and candidates, but the $\lceil \frac{1-\delta}{2}n + 1 \rceil$ voters previously located on b are now located almost at the halfway point between a and b with a slight inclination towards b such that their distance from b is $1 - \epsilon$ and their distance from a is 1 . The rest of the voters (of which there are $\lfloor \frac{1+\delta}{2}n - 1 \rfloor$) are located on a . Note that



instance \mathcal{I}_2 induces the same preference profile σ as \mathcal{I}_1 . Thus, if the real instance is \mathcal{I}_2 but **ALG** received the, now inaccurate, prediction $(\mathcal{V}, \mathcal{C}, d)$, along with σ , it would still choose b . Therefore, **ALG** is not β -robust, since here a is the optimal candidate, and the output b achieves distortion equal to:

$$\begin{aligned} \frac{\text{SC}(b, d)}{\text{SC}(a, d)} &= \frac{\lceil \frac{1-\delta}{2}n + 1 \rceil(1 - \epsilon) + \lfloor \frac{1+\delta}{2}n - 1 \rfloor(2 - \epsilon)}{\lceil \frac{1-\delta}{2}n + 1 \rceil \cdot 1} \\ &= \frac{n(1 - \epsilon) + \lfloor \frac{1+\delta}{2}n - 1 \rfloor \cdot 1}{\lceil \frac{1-\delta}{2}n + 1 \rceil} \geq \frac{n(1 - \epsilon) + (\frac{1+\delta}{2}n - 2)}{\frac{1-\delta}{2}n + 2} \\ &= \frac{(3 + \delta)\frac{n}{2} - 2 - \epsilon n}{\frac{1-\delta}{2}n + 2} = \frac{3 + \delta - \frac{4}{n} - 2\epsilon}{1 - \delta + \frac{4}{n}} = r(n, \epsilon) > \beta. \quad \square \end{aligned}$$

4.2 (p, q) -Algorithms with Prediction Regarding the Optimal Candidate

We now present a family of learning-augmented algorithms, termed LA_δ , which is parameterized by $\delta \in [0, 1]$. Instead of a prediction regarding the whole metric, these algorithms are provided only with a prediction regarding who the optimal candidate is, i.e., some $\hat{p} \in \mathcal{C}$. Note that this is a strictly weaker prediction: given an accurate prediction regarding the whole metric, one can readily infer a prediction regarding who the optimal candidate would be. More importantly, note that the algorithm needs to satisfy the consistency guarantee only on instances where $d \triangleright (\sigma, \hat{p})$. Therefore, if the prediction is regarding the whole metric, then the consistency guarantee binds only if the actual metric, d , is exactly the predicted metric. On the other hand, if the prediction is regarding the optimal candidate, there is an infinite number of different metrics d such that $c^*(d) = \hat{p}$, and the algorithm needs to achieve the consistency guarantee for all of them. Nevertheless, despite the weaker prediction that they are provided with, we prove that our algorithms achieve the optimal robustness-consistency trade-off that even algorithms with full-metric predictions can achieve.

Definition 4.2 (LA_δ). *Given a $\delta \in (0, 1]$, a predicted optimal candidate $\hat{c} \in \mathcal{C}$, and a preference profile σ , the class of LA_δ algorithms contains any $(p^{\text{uni}}, \hat{q})$ -algorithm with $\hat{q}(\hat{c}) = \frac{1-\delta}{1+\delta} \frac{p^{\text{lu}}(\hat{c}) + \frac{2\delta n}{1-\delta}}{n}$ as the weight of the predicted candidate and $\hat{q}(c) = \frac{1-\delta}{1+\delta} \frac{p^{\text{lu}}(c)}{n}$ the weight of any other candidate c .*

We refer to δ as the *confidence parameter* that the designer can choose, depending on how much she trusts the prediction (or, depending on the robustness-consistency trade-off that she prefers). The next theorem shows that for each δ , any LA_δ algorithm achieves a different Pareto optimal trade-off between robustness and consistency, matching the impossibility result of Theorem 4.1.

Theorem 4.3. For any $\delta \in [0, 1)$, LA_δ achieves $\frac{3-\delta}{1+\delta}$ -consistency and $\frac{3+\delta}{1-\delta}$ -robustness.

Proof. We first show that LA_δ has a distortion at most $\frac{3+\delta}{1-\delta}$ directly from the distortion bound we obtain from Section 3.1. In other words, any candidate that admits a fractional perfect matching with $(p^{\text{uni}}, \hat{q})$ has distortion $\frac{3+\delta}{1-\delta}$. This implies that the *robustness* of LA_δ algorithm is at most $\frac{3+\delta}{1-\delta}$. First note that given some δ instance σ , and \hat{q} defined by Definition 4.2, we have:

$$\mu(\sigma, \hat{q}) = \min_{c \in \mathcal{C}} \frac{\hat{q}(c) \cdot n}{\text{plu}(c)} = \frac{\frac{1-\delta}{1+\delta} \cdot \frac{\text{plu}(c)}{n} \cdot n}{\text{plu}(c)} = \frac{1-\delta}{1+\delta}.$$

From Theorem 3.4, the distortion of candidate a is at most $1 + \frac{2}{\mu}$ by letting $\mu = \frac{1-\delta}{1+\delta}$, we get that the distortion of $(p^{\text{uni}}, \hat{q})$ -algorithm is at most

$$1 + \frac{2}{\mu} = 1 + \frac{2(1+\delta)}{1-\delta} = \frac{3+\delta}{1-\delta}.$$

We now turn to the consistency of LA_δ , i.e., the case $\hat{c} = c^*$. Assume a is some candidate chosen by a LA_δ algorithm. The social cost of candidate a is:

$$\begin{aligned} \text{SC}(a, d) &= \sum_{v \in \mathcal{V}} d(v, a) \\ &= \sum_{v \in \mathcal{V}} n \cdot p^{\text{uni}}(v) \cdot d(v, a) && (\text{by Def. of } p^{\text{uni}}) \\ &= n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(v, a) && (\text{by Def. 2.2}) \end{aligned}$$

$$\begin{aligned}
&\leq n \sum_{v \in \mathcal{V}} \sum_{c:(v,c) \in E_a} w((v,c)) \cdot d(v,c) && \text{(by Def. 2.1)} \\
&\leq n \sum_{v \in \mathcal{V}} \sum_{c:(v,c) \in E_a} w((v,c)) \cdot (d(v, c^*) + d(c^*, c)) && \text{(triangle inequality)} \\
&\leq \text{SC}(c^*, d) + n \sum_{v \in \mathcal{V}} \sum_{c:(v,c) \in E_a} w((v,c)) \cdot d(c^*, c) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \sum_{v \in \mathcal{V}: (v,c) \in E_a} w((v,c)) \cdot d(c^*, c) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \hat{q}(c) \cdot d(c^*, c) && \text{(by Def. 2.2)} \\
&= \text{SC}(c^*, d) + n \cdot \frac{1-\delta}{1+\delta} \left(\sum_{c \neq \hat{c}} \frac{\text{plu}(c)}{n} \cdot d(c^*, c) + \frac{\text{plu}(\hat{c}) + \frac{2\delta n}{1-\delta}}{n} \cdot d(c^*, \hat{c}) \right) && \text{(by Def. 4.2)} \\
&= \text{SC}(c^*, d) + \frac{1-\delta}{1+\delta} \sum_{c \neq c^*} \text{plu}(c) \cdot d(c^*, c) && \text{(since } \hat{c} = c^* \text{)} \\
&= \text{SC}(c^*, d) + \frac{1-\delta}{1+\delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} d(c^*, c) \\
&\leq \text{SC}(c^*, d) + \frac{1-\delta}{1+\delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} (d(c, v) + d(c^*, v)) && \text{(triangle inequality)} \\
&\leq \text{SC}(c^*, d) + \frac{1-\delta}{1+\delta} \sum_c \sum_{v: \text{top}(v)=c} 2d(c^*, v) && \text{(since } c = \text{top}(v) \succ_v c^* \text{)} \\
&\leq \text{SC}(c^*, d) + \frac{2(1-\delta)}{1+\delta} \text{SC}(c^*, d) \leq \frac{3-\delta}{1+\delta} \text{SC}(c^*, d). && \square
\end{aligned}$$

4.3 Distortion Bounds Parameterized by Prediction Quality

Consistency and robustness bounds focus on two extreme cases: the case where the prediction is perfect (consistency) and the case where it can be arbitrarily bad (robustness). To develop a more refined understanding of the distortion that our family of LA_δ algorithms achieves as a function of the prediction quality, we now seek to prove bounds parameterized by the error of the prediction.

We first define a natural measure of error, which is proportional to the distance between the prediction, \hat{c} , and the optimal candidate, c^* , i.e., $d(\hat{c}, c^*)$. Note that using this distance alone as a measure of error would not accurately capture the quality of the prediction, since this does not measure how significant this distance is relative to the average cost that each voter would suffer in the optimal solution. For example, if $d(\hat{c}, c^*) = 100$, and the average distance in the optimal solution is 10000, then the prediction was quite accurate, whereas the prediction would be inaccurate if we scaled the same instance down so that the average distance is 1. Therefore, to accurately capture the relative quality of the prediction we quantify the error η of the prediction by normalizing its distance from c^* by the average optimal distance, $\text{SC}(c^*)/n$, i.e.,

$$\eta := \frac{nd(\hat{c}, c^*)}{\text{SC}(c^*)}$$

Our main result in this section shows that for each choice of $\delta \in [0, 1)$, our LA_δ algorithm guarantees distortion at most $\frac{3-\delta+2\delta\eta}{1+\delta}$ when the error of the prediction is η . This bound is equal

to our consistency bound when the prediction is correct, i.e., $\eta = 0$, and then grows linearly as a function of η , with a slope that is increasing in δ . This makes sense, as it captures the intuition that trusting the prediction more (i.e., higher values of δ) translates into a worse dependence on the error. Note, however, that the distortion never exceeds our robustness bound, so it gracefully transitions from the consistency bound to the robustness bound as a function of the error. Figure 3 demonstrates this bound for different values of δ . For example, as long as $\eta \leq 2$, the distortion is at most 3, for any value of δ . We defer the proof of the following Proposition to Appendix B due to space limitation.

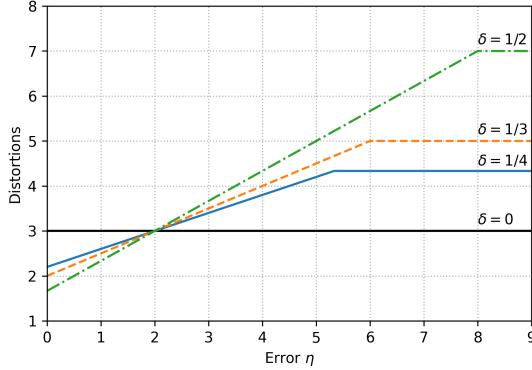


Figure 3: A plot of the distortion upper bound guaranteed by LA_δ as a function of the prediction error η for different values of $\delta \in [0, 1)$.

Proposition 4.4. *For any $\delta \in [0, 1)$, the distortion achieved by LA_δ on instances where the prediction \hat{c} has error $\eta = nd(\hat{c}, c^*)/\text{SC}(c^*)$ is at most*

$$\min \left\{ \frac{3 - \delta + 2\delta\eta}{1 + \delta}, \frac{3 + \delta}{1 - \delta} \right\}.$$

4.4 Bounds for Metric Spaces with Decisive Voters

We now refine the achieved distortion bounds even further, using the α -decisiveness framework of [5]. For $\alpha \in [0, 1]$, A voter v is α -decisive if $d(v, \text{top}(v)) \leq \alpha \cdot d(v, c)$, for all $c \neq \text{top}(v)$. In other words, the distance of voter v from her favorite candidate is at most α times her distance from any other candidate. We say an instance $(\mathcal{V}, \mathcal{C}, d)$ is α -decisive if every voter is α -decisive. Note that when $\alpha = 0$, the distance between every voter and her favorite candidate is 0. This captures the *peer selection* setting, in which the set of voters is the set of candidates, and each voter ranks herself first. Therefore, the positive results presented in the section for $\alpha = 0$ apply to this setting as well. Since the proofs of the decisiveness results are very similar to some of our earlier proofs, we defer them to Appendix C and directly provide the most general bound that is parameterized both by the decisiveness parameter α , and the error parameter η .

Proposition 4.5. *For any $\delta \in [0, 1)$ and $\alpha \in [0, 1]$, the distortion achieved by LA_δ on α -decisive instances where the prediction \hat{c} has error $\eta = nd(\hat{c}, c^*)/\text{SC}(c^*)$ is at most*

$$\frac{2 + \alpha - \alpha\delta}{1 + \delta} + \min \left\{ \frac{2\delta\eta}{1 + \delta}, \frac{8\delta}{(1 + \delta)(1 - \delta)} \right\}.$$

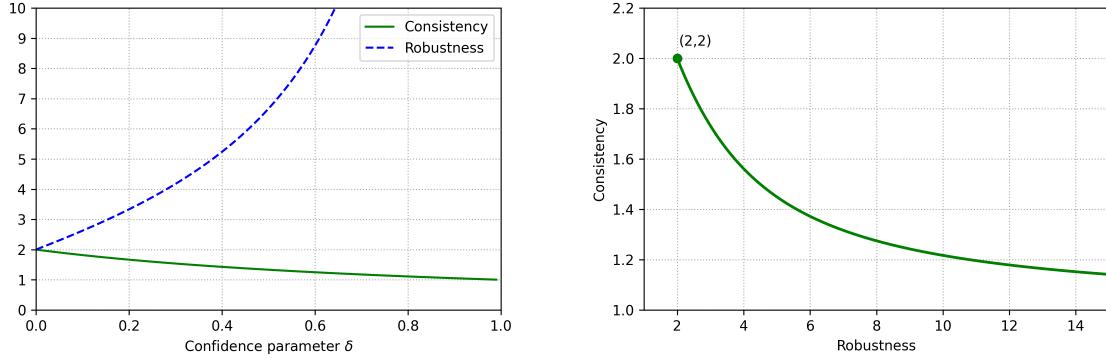


Figure 4: The analogue of Figure 1 for 0-decisive instances (peer selection setting). On the left, a plot showing the consistency and robustness bounds achieved by LA_δ as a function of the confidence parameter δ . On the right, a plot capturing the consistency-robustness pairs that can be achieved by LA_δ for any chosen $\delta \in [0, 1)$.

If we focus our attention on the interesting case where $\alpha = 0$, i.e., the peer selection case, then this implies the following bound.

Corollary 4.6. *For any $\delta \in [0, 1)$ and $\alpha = 0$, the distortion achieved by LA_δ on α -decisive instances where the prediction \hat{c} has error $\eta = nd(\hat{c}, c^*)/\text{SC}(c^*)$ is at most*

$$\frac{2}{1 + \delta} + \min \left\{ \frac{2\delta\eta}{1 + \delta}, \frac{8\delta}{(1 + \delta)(1 - \delta)} \right\}.$$

In the metric distortion problem with 0-decisive voters (the peer selection setting), the best known achievable distortion is 2. Our algorithm recovers this guarantee for $\delta = 0$. For $\delta > 0$, our algorithm achieves a consistency strictly better than 2, at the cost of higher robustness — Figure 4 illustrates the achieved consistency-robustness pairs obtained by varying δ . Note that as long as the value of η is at most 1, our algorithm ensures that the achieved distortion remains at most 2.

5 Conclusion and Future Work

Our contribution in this work is twofold: *i*) we establish a new distortion upper bound for candidates on the (p, q) -veto core for general p and q and *ii*) we initiate the study of distortion problems through the learning-augmented framework, leading to an optimal learning-augmented algorithm guided by a prediction for the optimal candidate.

An interesting avenue for future research involves a deeper exploration of the implications of our distortion bounds for the (p, q) -veto core. A particularly intriguing problem within this direction is identifying other well-motivated alternatives for the q vector. For example, Kizilkaya and Kempe [41] propose using k -plurality scores of candidates (the number of times they appear among the top k choices of a voter) to generate a q , which interpolates between q^{Plu} (for $k = 1$) and uniform q (for $k = m$). Another interesting direction is to examine the potential for alternative parameterizations of the distortion bounds. Identifying even more refined upper bounds as a function of p , q and Plu , or as a function of other parameters.

Our results also pave the way for revisiting many other interesting problems within the distortion literature using the learning-augmented framework (see [7] for a recent survey on the distortion

literature). Examples include the multi-winner variant of the metric distortion problem [17], the distortion of one-sided matching [28], as well as the utilitarian distortion setting with a single winner [50] or multiple winners [16]. Another natural direction is to apply this framework to randomized metric distortion, especially given the recent randomized algorithm achieving distortion better than 3 [23]. Can this be extended toward more general improvements on the robustness-consistency Pareto frontier achievable via randomization?

Acknowledgement

We would like to thank the anonymous reviewers for their detailed feedback and comments that helped significantly improve the exposition of this paper.

References

- [1] Ben Abramowitz, Elliot Anshelevich, and Wennan Zhu. Awareness of voter passion greatly improves the distortion of metric social choice. In *Proceedings of the The 15th Conference on Web and Internet Economics (WINE)*, pages 3–16, 2019.
- [2] Priyank Agrawal, Eric Balkanski, Vasilis Gkatzelis, Tingting Ou, and Xizhi Tan. Learning-augmented mechanism design: Leveraging predictions for facility location. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 497–528, 2022.
- [3] Noga Alon, Felix Fischer, Ariel Procaccia, and Moshe Tennenholtz. Sum of us: Strategyproof selection from the selectors. In *Proceedings of the 13th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 101–110, 2011.
- [4] Ioannis Anagnostides, Dimitris Fotakis, and Panagiotis Patsilinakos. Metric-distortion bounds under limited information. *Journal of Artificial Intelligence Research*, 74:1449–1483, 2022.
- [5] Elliot Anshelevich and John Postl. Randomized social choice functions under metric preferences. *Journal of Artificial Intelligence Research*, 58:797–827, 2017.
- [6] Elliot Anshelevich, Onkar Bhardwaj, Edith Elkind, John Postl, and Piotr Skowron. Approximating optimal social choice under metric preferences. *Artificial Intelligence*, 264:27–51, 2018. ISSN 0004-3702.
- [7] Elliot Anshelevich, Aris Filos-Ratsikas, Nisarg Shah, and Alexandros A. Voudouris. Distortion in social choice problems: The first 15 years and beyond. In *30th*, pages 4294–4301, 2021. Survey Track.
- [8] Antonios Antoniadis, Themis Gouleakis, Pieter Kleer, and Pavel Kolev. Secretary and online matching problems with machine learned advice. *Advances in Neural Information Processing Systems*, 33:7933–7944, 2020.
- [9] Yossi Azar, Debmalya Panigrahi, and Noam Touitou. Online graph algorithms with predictions. *Proceedings of the Thirty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, 2022.
- [10] Eric Balkanski, Vasilis Gkatzelis, and Xizhi Tan. Strategyproof scheduling with predictions. In *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2023.

- [11] Eric Balkanski, Vasilis Gkatzelis, Xizhi Tan, and Cherlin Zhu. Online mechanism design with predictions. In *Proceedings of the 25th ACM Conference on Economics and Computation (to appear)*, 2024.
- [12] Etienne Bamas, Andreas Maggiori, and Ola Svensson. The primal-dual method for learning augmented algorithms. *Advances in Neural Information Processing Systems*, 33:20083–20094, 2020.
- [13] Siddhartha Banerjee, Vasilis Gkatzelis, Artur Gorokh, and Billy Jin. Online nash social welfare maximization with predictions. In *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022*. SIAM, 2022.
- [14] Siddhartha Banerjee, Vasilis Gkatzelis, Safwan Hossain, Billy Jin, Evi Micha, and Nisarg Shah. Proportionally fair online allocation of public goods with predictions. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI*, pages 20–28, 2023.
- [15] Ioannis Caragiannis and Georgios Kalantzis. Randomized learning-augmented auctions with revenue guarantees. *CoRR*, abs/2401.13384, 2024.
- [16] Ioannis Caragiannis, Swaprava Nath, Ariel D. Procaccia, and Nisarg Shah. Subset selection via implicit utilitarian voting. *Journal of Artificial Intelligence Research*, 58:123–152, 2017.
- [17] Ioannis Caragiannis, Nisarg Shah, and Alexandros A. Voudouris. The metric distortion of multiwinner voting. *Artif. Intell.*, 313:103802, 2022.
- [18] Ching-Lueh Chang. Some results on approximate 1-median selection in metric spaces. *Theoretical Computer Science*, 426:1–12, 2012.
- [19] Ching-Lueh Chang. A lower bound for metric 1-median selection. *Journal of Computer and System Sciences*, 84:44–51, 2017.
- [20] Ching-Lueh Chang. Metric 1-median selection: Query complexity vs. approximation ratio. *ACM Transactions on Computation Theory (TOCT)*, 9(4):1–23, 2017.
- [21] Moses Charikar and Sudipto Guha. Improved combinatorial algorithms for the facility location and k-median problems. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pages 378–388. IEEE, 1999.
- [22] Moses Charikar and Prasanna Ramakrishnan. Metric distortion bounds for randomized social choice. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2986–3004. SIAM, 2022.
- [23] Moses Charikar, Kangning Wang, Prasanna Ramakrishnan, and Hongxun Wu. Breaking the metric voting distortion barrier. In David P. Woodruff, editor, *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1621–1640. SIAM, 2024.
- [24] Vincent Cohen-Addad, Michał Pilipczuk, and Marcin Pilipczuk. A polynomial-time approximation scheme for facility location on planar graphs. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 560–581. IEEE, 2019.
- [25] Paul Dütting, Silvio Lattanzi, Renato Paes Leme, and Sergei Vassilvitskii. Secretaries with advice. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 409–429, 2021.

- [26] James M Enelow and Melvin J Hinich. The spatial theory of voting. *Cambridge Books*, 1984.
- [27] Brandon Fain, Ashish Goel, Kamesh Munagala, and Nina Prabhu. Random dictators with a random referee: Constant sample complexity mechanisms for social choice. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pages 1893–1900, 2019.
- [28] Aris Filos-Ratsikas and Peter Bro Miltersen. Truthful approximations to range voting. In *Web and Internet Economics: 10th International Conference, WINE 2014, Beijing, China, December 14-17, 2014. Proceedings 10*, pages 175–188. Springer, 2014.
- [29] Dimitris Fotakis and Laurent Gourvès. On the distortion of single winner elections with aligned candidates. *Auton. Agents Multi Agent Syst.*, 36(2):37, 2022.
- [30] Vasilis Gkatzelis, Daniel Halpern, and Nisarg Shah. Resolving the optimal metric distortion conjecture. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1427–1438. IEEE, 2020.
- [31] Vasilis Gkatzelis, Kostas Kollias, Alkmini Sgouritsa, and Xizhi Tan. Improved price of anarchy via predictions. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 529–557, 2022.
- [32] Vasilis Gkatzelis, Mohamad Latifian, and Nisarg Shah. Best of both distortion worlds. In *The 24th ACM Conference on Economics and Computation*, 2023.
- [33] Stephen Gross, Elliot Anshelevich, and Lirong Xia. Vote until two of you agree: Mechanisms with small distortion and sample complexity. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 544–550, 2017.
- [34] Sungjin Im, Ravi Kumar, Mahshid Montazer Qaem, and Manish Purohit. Online knapsack with frequency predictions. *Advances in Neural Information Processing Systems*, 34, 2021.
- [35] Piotr Indyk. Sublinear time algorithms for metric space problems. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 428–434, 1999.
- [36] Gabriel Istrate and Cosmin Bonchis. Mechanism design with predictions for obnoxious facility location. *CoRR*, abs/2212.09521, 2022.
- [37] K. Jain, M. Mahdian, E. Markakis, A. Saberi, and V. V. Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *Journal of the ACM*, 50(6):795–824, 2003.
- [38] David Kempe. Communication, distortion, and randomness in metric voting. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2087–2094, 2020.
- [39] David Kempe. An analysis framework for metric voting based on LP duality. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2079–2086, 2020.
- [40] Fatih Erdem Kizilkaya and David Kempe. Plurality veto: A simple voting rule achieving optimal metric distortion. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 349–355, 2022.

[41] Fatih Erdem Kizilkaya and David Kempe. Generalized veto core and a practical voting rule with optimal metric distortion. In Kevin Leyton-Brown, Jason D. Hartline, and Larry Samuelson, editors, *Proceedings of the 24th ACM Conference on Economics and Computation, EC*, pages 913–936, 2023.

[42] Alexander Lindermayr and Nicole Megow. Alps. URL <https://algorithms-with-predictions.github.io/>.

[43] Pinyan Lu, Zongqi Wan, and Jialin Zhang. Competitive auctions with imperfect predictions. In *Proceedings of 25th ACM Conference on Economics and Computation (to appear)*, 2024.

[44] Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. In *International Conference on Machine Learning*, pages 3296–3305. PMLR, 2018.

[45] Michael Mitzenmacher and Sergei Vassilvitskii. *Algorithms with Predictions*, page 646–662. Cambridge University Press, 2021.

[46] Hervé Moulin. The proportional veto principle. *The Review of Economic Studies*, 48(3): 407–416, 1981. ISSN 00346527, 1467937X. URL <http://www.jstor.org/stable/2297154>.

[47] Kamesh Munagala and Kangning Wang. Improved metric distortion for deterministic social choice rules. In *Proceedings of the 2019 ACM Conference on Economics and Computation (EC)*, pages 245–262, 2019.

[48] Jannik Peters. A note on rules achieving optimal metric distortion. arXiv:2305.08667, 2023.

[49] Siddharth Prasad, Maria-Florina Balcan, and Tuomas Sandholm. Bicriteria multidimensional mechanism design with side information. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

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

[51] Haripriya Pulyassary and Chaitanya Swamy. On the randomized metric distortion conjecture. *CoRR*, abs/2111.08698, 2021. URL <https://arxiv.org/abs/2111.08698>.

[52] Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ml predictions. In *Advances in Neural Information Processing Systems*, pages 9661–9670, 2018.

[53] Chenyang Xu and Pinyan Lu. Mechanism design with predictions. In Luc De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 571–577. ijcai.org, 2022. doi: 10.24963/ijcai.2022/81. URL <https://doi.org/10.24963/ijcai.2022/81>.

A Missing Proofs from Section 3.2

Lemma 3.11. *Consider any candidate a in the (p, q) -veto core of σ for some arbitrary $p \in \Delta(\mathcal{V})$ and $q \in \Delta(\mathcal{C})$, then there exists some $q' \in \Delta(\mathcal{C})$ such that $G_{p^{uni}, q'}^\sigma(a)$ also admits a fractional perfect matching. In addition, if we let $\lambda = n \max_v p(v)$, we have*

$$\mu(\sigma, q') \geq \frac{\mu(\sigma, q)}{\lambda}.$$

Proof. We let $p_m = \max_v p(v)$, and for each voter v we define $\lambda_v = p_m - p(v)$. We note that $0 \leq \lambda_v \leq 1$. We construct the following q' vector as a function of the initial p, q as follows: for any candidate $c \in \mathcal{C}$, we define:

$$q'(c) = \frac{1}{\lambda} \cdot \left(q(c) + \sum_{v:(v,c) \in E_a} \frac{w((v,c)) \cdot \lambda_v}{p(v)} \right).$$

We first show that $q' \in \Delta(\mathcal{C})$. To this end, consider:

$$\begin{aligned} \sum_c q(c') &= \frac{1}{\lambda} \cdot \sum_c \left(q(c) + \sum_{v:(v,c) \in E_a} \frac{w((v,c)) \cdot \lambda_v}{p(v)} \right) \\ &= \frac{1}{\lambda} \left(\sum_c q(c) + \sum_c \sum_{v:(v,c) \in E_a} \frac{w((v,c)) \cdot \lambda_v}{p(v)} \right) \\ &= \frac{1}{\lambda} \cdot \left(1 + \sum_c \sum_{v:(v,c) \in E_a} \frac{w((v,c)) \cdot \lambda_v}{p(v)} \right) && (\text{since } q \in \Delta(\mathcal{C})) \\ &= \frac{1}{\lambda} \cdot \left(1 + \sum_v \sum_{c:(v,c) \in E_a} \frac{w((v,c)) \cdot \lambda_v}{p(v)} \right) \\ &= \frac{1}{\lambda} \cdot \left(1 + \sum_v \lambda_v \right) && (\text{by Def. 2.2}) \\ &= \frac{1}{\lambda} \cdot (\lambda) = 1 && (\text{by Def. of } \lambda) \end{aligned}$$

We now show that $G_{p^{\text{uni}}, q'}^\sigma(a)$ admits a fractional perfect matching. Consider the following weight function w' : for any $(v, c) \in E_a$, we let:

$$w'((v, c)) = \frac{1}{\lambda} \cdot \left(1 + \frac{\lambda_v}{p(v)} \right) \cdot w((v, c)).$$

First, consider any candidate c :

$$\begin{aligned} \sum_{v:(v,c) \in E_a} w'((v, c)) &= \frac{1}{\lambda} \cdot \sum_{v:(v,c) \in E_a} \left(1 + \frac{\lambda_v}{p(v)} \right) \cdot w((v, c)) \\ &= \frac{1}{\lambda} \cdot \sum_{v:(v,c) \in E_a} \left(w((v, c)) + \frac{w((v,c)) \cdot \lambda_v}{p(v)} \right) \\ &= \frac{1}{\lambda} \cdot \left(q(c) + \sum_{v:(v,c) \in E_a} \frac{w((v,c)) \cdot \lambda_v}{p(v)} \right) && (\text{by Def. 2.2}) \\ &= q'(c) \end{aligned}$$

Now, consider any voter v :

$$\begin{aligned}
\sum_{c:(v,c) \in E_a} w'((v, c)) &= \frac{1}{\lambda} \cdot \left(1 + \frac{\lambda_v}{p(v)}\right) \cdot \sum_{c \in \mathcal{C}} w((v, c)) \\
&= \frac{1}{\lambda} \cdot \left(1 + \frac{\lambda_v}{p(v)}\right) \cdot p(v) \\
&= \frac{p(v) + \lambda_v}{\lambda} \\
&= \frac{p_m}{\sum_v (p(v) + \lambda_v)} \quad (\text{by Def. of } \lambda \text{ and } p \in \Delta(\mathcal{V})) \\
&= \frac{p_m}{n \cdot p_m} = \frac{1}{n} = p^{\text{uni}}(v).
\end{aligned}$$

Finally, we consider the change in μ value from q to q' . First, since $\lambda_v \geq 0$, we have

$$q'(c) = \frac{1}{\lambda} \cdot \left(q(c) + \sum_{v:(v,c) \in E_a} \frac{w((v, c)) \cdot \lambda_v}{p(v)} \right) \geq \frac{q(c)}{\lambda}.$$

Therefore, for any candidate c , we have:

$$\frac{q'(c) \cdot n}{\text{plu}(c)} \geq \frac{1}{\lambda} \cdot \frac{q(c) \cdot n}{\text{plu}(c)}.$$

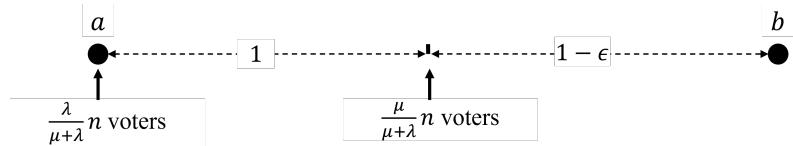
Since the pointwise change of any candidate c is bounded by $\frac{1}{\lambda}$, we therefore have

$$\mu(\sigma, q') = \min_{c \in \mathcal{C}} \frac{q'(c) \cdot n}{\text{plu}(c)} \geq \min_{c \in \mathcal{C}} \frac{1}{\lambda} \cdot \frac{q(c) \cdot n}{\text{plu}(c)} = \frac{\mu(\sigma, q)}{\lambda}. \quad \square$$

Theorem 3.13. Given any $\mu \in (0, 1]$ and $\lambda \in [1, n]$, there exists a preference profile σ and weight vectors $q \in \Delta(\mathcal{C})$ with $\mu = \min_{c \in \mathcal{C}} \frac{q(c) \cdot n}{\text{plu}(c)}$ and $p \in \Delta(\mathcal{V})$ with $\lambda = n \max_v p(v)$, such that for any $\epsilon > 0$ there is a metric $d \triangleright \sigma$ and a candidate a in the (p, q) -veto core of σ with

$$\text{SC}(a, d) \geq \left(1 + \frac{2\lambda}{\mu} - \epsilon\right) \text{SC}(c^*, d).$$

Proof. Consider the following instance $(\mathcal{V}, \mathcal{C}, d)$ consisting of two candidates a, b below. We first note



that in this instance, $\text{plu}(a) = \frac{\lambda}{\mu+\lambda}n$ and $\text{plu}(b) = \frac{\mu}{\mu+\lambda}n$. Now consider q such that $q(a) = \frac{\mu\lambda}{\mu+\lambda}$ and $q(b) = \frac{\mu+\lambda-\mu\lambda}{\mu+\lambda}$, it is easy to verify that $\min_c \frac{q(c)n}{\text{plu}(c)} = \frac{q(a)n}{\text{plu}(a)} = \mu$. We now partition the voters into V_1 and V_2 , where V_1 is the set of voters co-located with candidate a and V_2 is the set of voters co-located in the midpoint. For any $v \in V_1$, we set $p(v) = \frac{1}{n} - \frac{(\lambda-1)\mu}{\lambda n}$, and for any voter $v' \in V_2$, we set $p(v') = \frac{1}{n} + \frac{\lambda-1}{n}$. Note that $\max_v (p(v)) = \frac{1}{n} + \frac{\lambda-1}{n}$ and $n \cdot \max_v (p(v)) = 1 + \lambda - 1 = \lambda$.

$$\begin{aligned} v \in V_1 : \quad a \succ b \quad |V_1| = \frac{\lambda}{\mu+\lambda} \\ v \in V_2 : \quad b \succ a \quad |V_2| = \frac{\mu}{\mu+\lambda} \end{aligned}$$

(a) Preference Profile σ

$$\begin{aligned} p(v) = \frac{1}{n} - \frac{(\lambda-1)\mu}{\lambda n} \quad (V_1) \quad q(a) = \frac{\mu\lambda}{\mu+\lambda} \\ p(v) = \frac{1}{n} + \frac{\lambda-1}{n} \quad (V_2) \quad q(b) = \frac{\mu+\lambda-\mu\lambda}{\mu+\lambda} \quad (b) \end{aligned}$$

(b) Domination Graph $G_{p,q}^\sigma(b)$

Figure 5: the preference profile and the domination graph induced from the instance and the choice of p, q .

We now argue that candidate b , given the p, q vector defined above has a fractional perfect matching. Consider the graph $G_{p,q}^\sigma(b)$, note that, for voters $v \in V_1$ they have only edges to b , whereas the voters in V_2 has edges to both candidate a, b . Consider a simple weight function w where $w((v, b)) = p(v) = \frac{1}{n} - \frac{(\lambda-1)\mu}{\lambda n}$ and $w((v, a)) = 0$ for all $v \in V_1$ and $w((v', a)) = \frac{1}{n} + \frac{\lambda-1}{n}$ and $w((v, b)) = 0$ for all $v' \in V_2$. We first note that for each voter v , $w((v, a)) + w((v, b)) = p(v)$. Now consider the candidates, we have:

$$\sum_{v:(v,b) \in E_b} w((v, b)) = \left(\frac{1}{n} - \frac{(\lambda-1)\mu}{\lambda n} \right) \cdot |V_1| = \left(\frac{1}{n} - \frac{(\lambda-1)\mu}{\lambda n} \right) \cdot \frac{\lambda}{\mu+\lambda} n = \frac{\mu+\lambda-\mu\lambda}{\mu+\lambda} = q(b).$$

$$\sum_{v:(v,a) \in E_b} w((v, a)) = \left(\frac{1}{n} + \frac{\lambda-1}{n} \right) \cdot |V_2| = \left(\frac{1}{n} + \frac{\lambda-1}{n} \right) \cdot \frac{\mu}{\mu+\lambda} n = \frac{\mu\lambda}{\mu+\lambda} = q(a).$$

We therefore conclude that $G_{p,q}^\sigma(b)$ admits a fractional perfect matching. We now compute the distortion of candidate b . To this end, we first compute the optimal social cost of the instance, note that the optimal candidate in this case is candidate a , we have:

$$\text{SC}(a, d) = \frac{\mu}{\mu+\lambda} n \cdot 1 + \frac{1}{1+\mu} n \cdot 0 = \frac{\mu}{\mu+\lambda} n.$$

The cost of candidate b is:

$$\begin{aligned} \text{SC}(b, d) &= \frac{\mu}{\mu+\lambda} n \cdot (1-\epsilon) + \frac{\lambda}{\mu+\lambda} n \cdot (2-\epsilon) \\ &= \frac{\mu+2\lambda}{\mu+\lambda} n - \epsilon n = \left(1 + \frac{2\lambda}{\mu} - \frac{\epsilon(\mu+\lambda)}{\mu} \right) \text{SC}(a, d) \end{aligned}$$

the distortion converges to $1 + \frac{2\lambda}{\mu}$ as ϵ converges to 0. \square

B Missing Proofs from Section 4.3

Proposition 4.4. *For any $\delta \in [0, 1)$, the distortion achieved by LA_δ on instances where the prediction \hat{c} has error $\eta = nd(\hat{c}, c^*)/\text{SC}(c^*)$ is at most*

$$\min \left\{ \frac{3-\delta+2\delta\eta}{1+\delta}, \frac{3+\delta}{1-\delta} \right\}.$$

Proof. For any $\delta \in [0, 1)$, assume that a is the candidate output by a given execution of LA_δ , when provided with a prediction \hat{c} with error $\eta = nd(\hat{c}, c^*)/\text{SC}(c^*, d)$. Theorem 4.3 already shows that the worst-case distortion of LA_δ is at most $\frac{3+\delta}{1-\delta}$ for any δ , so we just need to prove that it is also at most $\frac{3-\delta+2\delta\eta}{1+\delta}$.

The social cost of a can be upper bounded as follows:

$$\begin{aligned}
\text{SC}(a, d) &= \sum_{v \in \mathcal{V}} d(v, a) \\
&= \sum_{v \in \mathcal{V}} n \cdot p^{\text{uni}}(v) \cdot d(v, a) && \text{(by Def. of } p^{\text{uni}}\text{)} \\
&= n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(v, a) && \text{(by Def. 2.2)} \\
&\leq n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(v, c) && \text{(by Def. 2.1)} \\
&\leq n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot (d(v, c^*) + d(c^*, c)) && \text{(triangle inequality)} \\
&\leq \text{SC}(c^*, d) + n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \sum_{v \in \mathcal{V}: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \hat{q}(c) \cdot d(c^*, c) && \text{(by Def. 2.2)} \\
&= \text{SC}(c^*, d) + n \cdot \frac{1 - \delta}{1 + \delta} \left(\sum_{c \neq \hat{c}} \frac{\text{plu}(c)}{n} \cdot d(c^*, c) + \frac{\text{plu}(\hat{c}) + \frac{2\delta n}{1 - \delta}}{n} \cdot d(c^*, \hat{c}) \right) && \text{(by Def. 4.2)} \\
&= \text{SC}(c^*, d) + \frac{2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \neq c^*} \text{plu}(c) \cdot d(c^*, c) && \text{(by } d(c^*, \hat{c}) = \eta \cdot \text{SC}(c^*, d)/n\text{)} \\
&= \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} d(c^*, c) \\
&\leq \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} (d(c, v) + d(c^*, v)) && \text{(triangle inequality)} \\
&\leq \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_c \sum_{v: \text{top}(v)=c} 2d(c^*, v) && \text{(since } c = \text{top}(v) \succ_v c^*\text{)} \\
&\leq \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{2(1 - \delta)}{1 + \delta} \text{SC}(c^*, d) \leq \left(\frac{3 - \delta + 2\delta\eta}{1 + \delta} \right) \text{SC}(c^*, d). \square
\end{aligned}$$

C Missing Proofs from Section 4.4

C.1 Consistency Bound for α -Decisive Instances

Proposition C.1. *For any $\delta \in [0, 1)$ and $\alpha \in [0, 1]$, assume that a is the candidate output by a given execution of LA_δ on an α -decisive instance when provided with the correct prediction $\hat{c} = c^*$. Then we have*

$$\text{SC}(a, d) \leq \frac{2 + \alpha - \alpha\delta}{1 + \delta} \cdot \text{SC}(c^*, d).$$

Proof. The social cost of the returned candidate a can be upper bounded as follows:

$$\begin{aligned}
\text{SC}(a, d) &= \sum_{v \in \mathcal{V}} d(v, a) \\
&= \sum_{v \in \mathcal{V}} n \cdot p^{\text{uni}}(v) \cdot d(v, a) && (\text{by Def. of } p^{\text{uni}}) \\
&= n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(v, a) && (\text{by Def. 2.2}) \\
&\leq n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(v, c) && (\text{by Def. 2.1}) \\
&\leq n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot (d(v, c^*) + d(c^*, c)) && (\text{triangle inequality}) \\
&\leq \text{SC}(c^*, d) + n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \sum_{v \in \mathcal{V}: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \hat{q}(c) \cdot d(c^*, c) && (\text{by Def. 2.2}) \\
&= \text{SC}(c^*, d) + n \cdot \frac{1 - \delta}{1 + \delta} \left(\sum_{c \neq \hat{c}} \frac{\text{plu}(c)}{n} \cdot d(c^*, c) + \frac{\text{plu}(\hat{c}) + \frac{2\delta n}{1 - \delta}}{n} \cdot d(c^*, \hat{c}) \right) && (\text{by Def. 4.2}) \\
&= \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \neq c^*} \text{plu}(c) \cdot d(c^*, c) && (\text{since } \hat{c} = c^*) \\
&= \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} d(c^*, c) \\
&\leq \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} (d(c, v) + d(c^*, v)) && (\text{triangle inequality}) \\
&\leq \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} (\alpha d(c^*, v) + d(c^*, v)) && (\alpha\text{-decisiveness}) \\
&\leq \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_c \sum_{v: \text{top}(v)=c} (1 + \alpha) d(c^*, v) \\
&\leq \text{SC}(c^*, d) + \frac{(1 + \alpha)(1 - \delta)}{1 + \delta} \text{SC}(c^*, d) \leq \frac{2 + \alpha - \alpha\delta}{1 + \delta} \text{SC}(c^*, d). && \square
\end{aligned}$$

C.2 Robustness Bound for α -Decisive Instances

Proposition C.2. *For any $\delta \in [0, 1)$ and $\alpha \in [0, 1]$, assume that \hat{c} is the candidate output by a given execution of LA_δ on an α -decisive instance, i.e., the algorithm returns the prediction. Then we have*

$$\text{SC}(\hat{c}, d) \leq \left(\frac{2 + \alpha - \alpha\delta}{1 + \delta} + \frac{8\delta}{(1 + \delta)(1 - \delta)} \right) \cdot \text{SC}(c^*, d).$$

Proof. We first note that by Definition 3.1, any candidates output by LA_δ is a member of (p, q) -veto core for the corresponding p, q . The social cost of the returned candidate a can then be upper

bounded as follows given a α -decisiveness instance. Note that the proof is very similar with the proof of Theorem 3.4.

$$\begin{aligned}
\mathsf{SC}(a, d) &= \sum_{v \in \mathcal{V}} d(v, a) \\
&= \sum_{v \in \mathcal{V}} n \cdot p(v) \cdot d(v, a) && \text{(since } p \text{ is uniform)} \\
&= n \sum_{v \in \mathcal{V}} \sum_{c \in \mathcal{C}: (v, c) \in E_a} w((v, c)) \cdot d(v, a) && \text{(by Def. 2.2)} \\
&= n \sum_{c \in \mathcal{C}} \sum_{v: (v, c) \in E_a} \left(\frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} + \left(1 - \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \right) \right) \cdot w((v, c)) \cdot d(v, a) \\
&= n \sum_{c \in \mathcal{C}} \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(v, a) + n \sum_{c \in \mathcal{C}} \left(1 - \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \right) \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(v, a) \\
&\leq n \sum_{c \in \mathcal{C}} \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(v, c) && \text{(by } a \succsim_v c) \\
&\quad + n \sum_{c \in \mathcal{C}} \left(1 - \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \right) \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(v, a) \\
&\leq n \sum_{c \in \mathcal{C}} \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \sum_{v: (v, c) \in E_a} w((v, c)) \cdot (d(v, c^*) + d(c^*, c)) \\
&\quad + n \sum_{c \in \mathcal{C}} \left(1 - \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \right) \sum_{v: (v, c) \in E_a} w((v, c)) \cdot (d(v, c^*) + d(c^*, a)) && \text{(triangle inequality)} \\
&= n \sum_{c \in \mathcal{C}} \left(\frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} + 1 - \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \right) \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(v, c^*) \\
&\quad + n \sum_{c \in \mathcal{C}} \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) \\
&\quad + n \sum_{c \in \mathcal{C}} \left(1 - \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \right) \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(c^*, a) \\
&= n \sum_{v \in \mathcal{V}} \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(v, c^*) \\
&\quad + n \sum_{c \in \mathcal{C}} \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) && \text{(rearranging the sum)} \\
&\quad + n \sum_{c \in \mathcal{C}} \left(1 - \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \right) \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(c^*, a) \\
&= \mathsf{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) && \text{(by Def. 2.2)} \\
&\quad + n \sum_{c \in \mathcal{C}} \left(1 - \frac{\mu \cdot \mathsf{plu}(c)}{q(c)n} \right) \sum_{v: (v, c) \in E_a} w((v, c)) \cdot d(c^*, a)
\end{aligned}$$

$$\begin{aligned}
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \frac{\mu \cdot \text{plu}(c)}{q(c)n} \cdot q(c) \cdot d(c^*, c) \\
&\quad + n \sum_{c \in \mathcal{C}} \left(1 - \frac{\mu \cdot \text{plu}(c)}{q(c)n}\right) q(c) \cdot d(c^*, a) \quad (\text{by Def. 2.2}) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} \frac{\mu}{n} \cdot d(c^*, c) + n(1 - \mu) \cdot d(c^*, a) \\
&= \text{SC}(c^*, d) + \mu \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v)=c} (d(c^*, v) + d(v, c)) + n(1 - \mu) \cdot d(c^*, a) \quad (\text{triangle inequality}) \\
&\leq \text{SC}(c^*, d) + \mu \sum_c \sum_{v: \text{top}(v)=c} (1 + \alpha) d(c^*, v) + n(1 - \mu) d(c^*, a) \quad (\alpha\text{-decisiveness}) \\
&\leq \text{SC}(c^*, d) + (1 + \alpha)\mu \cdot \text{SC}(c^*, d) + n(1 - \mu) \cdot \frac{2\text{SC}(c^*, d)(1 + \mu)}{\mu n} \quad (\text{by Lemma 3.10}) \\
&\leq \left(1 + (1 + \alpha)\mu + \frac{2(1 - \mu)(1 + \mu)}{\mu}\right) \text{SC}(c^*, d) \\
&= \left(\frac{2 + \alpha - \alpha\delta}{1 + \delta} + \frac{8\delta}{(1 + \delta)(1 - \delta)}\right) \cdot \text{SC}(c^*, d). \quad (\text{since } \mu = \frac{1 - \delta}{1 + \delta})
\end{aligned}$$

This concludes the proof. \square

C.3 Bounds Parameterized by Prediction Quality for α -Decisive Instances

Proposition C.3. *For any $\delta \in [0, 1)$ and $\alpha \in [0, 1]$, the distortion achieved by LA_δ on α -decisive instances where the prediction \hat{c} has error $\eta = nd(\hat{c}, c^*)/\text{SC}(c^*, d)$ is at most*

$$\frac{2 + \alpha - \alpha\delta}{1 + \delta} + \min \left\{ \frac{2\delta\eta}{1 + \delta}, \frac{8\delta}{(1 + \delta)(1 - \delta)} \right\}.$$

Proof. For any $\delta \in [0, 1)$, assume that a is the candidate output by a given execution of LA_δ on an α -decisive instance when provided with a prediction \hat{c} with error $\eta = nd(\hat{c}, c^*)/\text{SC}(c^*, d)$. Proposition C.2 already show that the worst-case distortion of LA_δ is at most $\left(\frac{2 + \alpha - \alpha\delta}{1 - \delta} + \frac{8\delta}{(1 + \delta)(1 - \delta)}\right)$ for any δ , so we just need to prove that it is also at most $\frac{2 + \alpha - \alpha\delta + 2\delta\eta}{1 + \delta}$. Note that if $\hat{c} = c^*$ then $\eta = 0$ and the claim follows from Proposition C.1. We thus assume for the rest of the proof that $\hat{c} \neq c^*$.

The social cost of a can be upper bounded as follows:

$$\begin{aligned}
\text{SC}(a, d) &= \sum_{v \in \mathcal{V}} d(v, a) \\
&= \sum_{v \in \mathcal{V}} n \cdot p^{\text{uni}}(v) \cdot d(v, a) \quad (\text{by Def. of } p^{\text{uni}}) \\
&= n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(v, a) \quad (\text{by Def. 2.2}) \\
&\leq n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(v, c) \quad (\text{by Def. 2.1}) \\
&\leq n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot (d(v, c^*) + d(c^*, c)) \quad (\text{triangle inequality})
\end{aligned}$$

$$\begin{aligned}
&\leq \text{SC}(c^*, d) + n \sum_{v \in \mathcal{V}} \sum_{c: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \sum_{v \in \mathcal{V}: (v, c) \in E_a} w((v, c)) \cdot d(c^*, c) \\
&= \text{SC}(c^*, d) + n \sum_{c \in \mathcal{C}} \hat{q}(c) \cdot d(c^*, c) \quad (\text{by Def. 2.2}) \\
&= \text{SC}(c^*, d) + n \cdot \frac{1 - \delta}{1 + \delta} \left(\sum_{c \neq \hat{c}} \frac{\text{plu}(c)}{n} \cdot d(c^*, c) + \frac{\text{plu}(\hat{c}) + \frac{2\delta n}{1 - \delta}}{n} \cdot d(c^*, \hat{c}) \right) \quad (\text{by Def. 4.2}) \\
&= \text{SC}(c^*, d) + \frac{2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \neq c^*} \text{plu}(c) \cdot d(c^*, c) \quad (\text{by } d(c^*, \hat{c}) = \eta \cdot \text{SC}(c^*, d)/n) \\
&= \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v) = c} d(c^*, c) \\
&\leq \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v) = c} (d(c, v) + d(c^*, v)) \quad (\text{triangle inequality}) \\
&\leq \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_{c \in \mathcal{C}} \sum_{v: \text{top}(v) = c} (\alpha d(c^*, v) + d(c^*, v)) \quad (\alpha\text{-decisiveness}) \\
&\leq \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{1 - \delta}{1 + \delta} \sum_c \sum_{v: \text{top}(v) = c} (1 + \alpha) d(c^*, v) \quad (\text{since } c = \text{top}(v) \succsim_v c^*) \\
&\leq \frac{1 + \delta + 2\delta\eta}{1 + \delta} \text{SC}(c^*, d) + \frac{(1 + \alpha)(1 - \delta)}{1 + \delta} \text{SC}(c^*, d) \leq \left(\frac{2 + \alpha - \alpha\delta + 2\delta\eta}{1 + \delta} \right) \text{SC}(c^*, d). \square
\end{aligned}$$