Finding Possible and Necessary Winners in Spatial Voting with Partial Information

Aviram Imber, Jonas Israel, Markus Brill, Hadas Shachnai, Benny Kimelfeld

Abstract

We consider spatial voting where candidates are located in the Euclidean d-dimensional space and each voter ranks candidates based on their distance from the voter's ideal point. We explore the case where information about the location of voters' ideal points is incomplete: for each dimension, we are given an interval of possible values. We study the computational complexity of finding the possible and necessary winners for positional scoring rules. Our results show that we retain tractable cases of the classic model where voters have partial-order preferences. Moreover, we show that there are positional scoring rules under which the possible-winner problem is intractable for partial orders, but tractable in the one-dimension spatial setting (while intractable in higher fixed number of dimensions).

1 Introduction

In the spatial model of voting [11, 19], both candidates and voters are associated with points in the d-dimensional Euclidean space $\mathbb{R}^d$. It is assumed that the locations of voters correspond to the voters' "ideal points" and that each voter's preferences over the candidates can be inferred from the Euclidean distance between the candidate and the voter's ideal point. For example, the location of candidates/voters in $\mathbb{R}^d$ could reflect the stance or opinion of the candidate/voter regarding d different *issues* that are relevant for the election. In the social choice literature, preferences with this structure are referred to as (d-)*Euclidean preferences* [5, 10].

We consider a setting in which only partial information about the preferences of voters is available. In this setting, the exact preference order of a voter is unknown, but assumed to come from a known space of possible preference orders. Each combination of possible preference orders is a possible voting profile that may result in different sets of winners (given a fixed voting rule). Natural computational tasks that arise in such scenarios ask about the *possible winners* (who win in at least one possible profile) and the *necessary winners* (who win in every possible profile) [16]. A prominent manifestation of this idea is the seminal framework of Konczak and Lang [15], in which voter preferences are specified as partial orders, and a possible profile is obtained by extending each partial order into a total preference order. A thorough picture of the complexity of the possible and necessary winner problems has been established in a series of studies [2, 3, 24]. For example, under every positional scoring rule in the setting of partial orders, the necessary winners can be found in polynomial time, yet it is NP-complete to decide whether a candidate is a possible winner (assuming a regularity condition that the rule is "pure"), except for the tractable cases of the plurality and veto rules.

In this work, we study the complexity of the computational problems $\mathsf{PW}\langle d\rangle$ and $\mathsf{NW}\langle d\rangle$, where the goal is to find the possible and necessary winners, respectively, when we have incomplete information about voters' ideal points in a spatial voting model with d dimensions. More precisely, instead of ideal points of the voters, we are given—for each voter and each dimension—an interval of possible values for the voter's ideal point. Hence, each voter is associated with a space of possible ideal points. Different points from this space may induce different preference orders over the candidates (whose locations are assumed to be known

Problem	plurality & veto	two-valued rules	other positional scoring rules
$\mathsf{NW}\langle d\rangle$	P [Thm. 2]	P [Thm. 2]	P [Thm. 2]
$\mathsf{PW}\langle 1\rangle$	P [Thm. 6]	P [Thm. 3]	P for weighted veto [Thm. 4] and for $F(k,t)$ whenever $k > t$ [Thm. 5]
$\mathsf{PW}\langle d\rangle$	P [Thm. 6]	NP-complete for k-approval for all $d \geq 2, k \geq 3$ [Thm. 7]	?

Table 1: Complexity results for computing necessary and possible winners.

precisely). Hence, we get a mechanism for defining a space of possible total orders that is different[1] from the classical partial-order setting [15]. We refer to our setting as *partial spatial voting*.

We focus on the class of positional scoring rules and compare the computational complexity of possible and necessary winner problems to the classic model with partial orders. In particular, we are interested in the following questions: (1) Is the necessary-winner problem still tractable for all positional scoring rules? (2) Is the possible-winner problem still tractable for plurality and veto? (3) Are there positional scoring rules where the possible-winner problem is tractable for partial spatial voting but not for partial orders? We answer all three questions positively. Our results are summarized in Table 1.

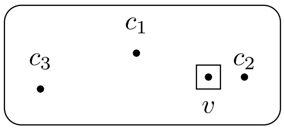
Related work. Spatial voting in one dimension is intuitively similar to assuming *single-peaked preferences* [4, 1], where the order should behave like a specific location in a line of positions. However, there are considerable differences, since single-peaked preferences do not impose any restrictions on the comparison between candidates on different sides of the voter. Walsh [22] showed hardness results in the setting where preferences are assumed to be single-peaked (but not necessarily 1-Euclidean) for STV and polynomial time results for Condorcet rules.

Bogomolnaia and Laslier [5] have shown that *every* (complete) preference profile can be represented in the spatial model, by choosing the dimension d to be sufficiently large. Given a preference profile, it can be efficiently checked whether the profile can be represented as a one-dimensional spatial model [6, 14, 9]. However, for higher dimensions, the problem is computationally intractable [20].

The problems we consider here relate also to control and manipulation problems that involve reasoning about a space of possibilities of profiles. Lu et al. [17] study control where a party can select a subset of issues to focus on. Estornell et al. [12] study manipulation of spatial voting where the issues are weighted and a malicious attacker can change the weights. Wu et al. [23] study manipulation where the adversary can change the position of a candidate by some quantity.

Organization. The remainder of the paper is organized as follows. After preliminary definitions in Section 2, we define partial spatial voting in Section 3. In Section 4 and Section 5 we study the problems of necessary winners and possible winners, respectively. We conclude in Section 6. Some of the proofs (omitted for lack of space) can be found in the Appendix.

[1] In Section 3, we show that the two mechanisms are indeed incomparable.



$\Rightarrow \quad c_2 \succ_v c_1 \succ_v c_3$

Figure 1: An example of a 2-dimenional spatial voting profile. There are three candidates $C = \{c_1, c_2, c_3\}$ and a single voter v. The ranking of v is (c_2, c_1, c_3).

2 Preliminaries

We first give preliminary concepts and notations that we use throughout the paper. For a natural number n, we let $[n]$ denote the set $\{1, \dots, n\}$.

2.1 Voting Profiles

Let $C = \{c_1, \dots, c_m\}$ be a set of *candidates* and $V = \{v_1, \dots, v_n\}$ a set of *voters*. We assume that $m = |C| \geq 2$. A *ranking profile* $\mathbf{R} = (R_1, \dots, R_n)$ consists of n linear orders on C, such that for each $i \in [n]$, R_i represents the preference order of voter v_i.

One way of obtaining a ranking profile is through *spatial voting* [11]. We associate each candidate with a d-dimensional vector corresponding to its positions (opinions) on issues, denoted as $\mathbf{c}_i = \langle c_{i,1}, \dots, c_{i,d} \rangle \in \mathbb{R}^d$. A *spatial voting profile* $\mathbf{T} = (T_1, \dots, T_n)$ consists of a vector $T_j = \langle T_{j,1}, \dots, T_{j,d} \rangle$ for each voter v_j, which represents its positions on the issues. Given a spatial profile $\mathbf{T}$, we can construct a ranking profile $\mathbf{R_T} = (R_{T_1}, \dots, R_{T_n})$ where each voter v_j ranks candidates in C according to their Euclidean distance from v_j, $||T_j - \mathbf{c}_i||_2$. The closest candidate is ranked first, and the farthest is ranked in position m in v_j's preferences. We break ties by a linear order on the candidates, which is given as part of the input for each voter. An example of a spatial voting profile and its associated ranking profile is illustrated in Figure 1. Throughout the paper, we identify voters with their ideal points in $\mathbb{R}^d$, and we use the terms *dimension* and *issue* interchangeably.

2.2 Voting Rules

A *voting rule* is a function that maps a ranking profile to a nonempty set of *winners*. In this paper, we focus on positional scoring rules.

Formally, a *positional scoring rule* r is a series $\{\vec{s}_m\}_{m \geq 2}$ of m-dimensional score vectors $\vec{s}_m = (\vec{s}_m(1), \dots, \vec{s}_m(m))$ of natural numbers, where $\vec{s}_m(1) \geq \cdots \geq \vec{s}_m(m)$ and $\vec{s}_m(1) > \vec{s}_m(m)$. Some examples of positional scoring rules include the *plurality* rule $(1, 0, \dots, 0)$, the k-*approval* rule $(1, \dots, 1, 0, \dots, 0)$ that begins with k ones, the *veto* rule $(1, \dots, 1, 0)$, the k-*veto* rule that ends with k zeros, and the *Borda* rule $(m-1, m-2, \dots, 0)$.

Given a ranking profile $\mathbf{R} = (R_1, \dots, R_n)$ and a positional scoring rule r, the score $s_r(R_i, c)$ that the voter v_i contributes to the candidate c is $\vec{s}_m(j)$, where j is the position of c in R_i. The score of c in $\mathbf{R}$ is $s_r(\mathbf{R}, c) = \sum_{i=1}^{n} s_r(R_i, c)$, which we may denote as $s(\mathbf{R}, c)$ if r is clear from context. A candidate c is a *winner* if $s_r(\mathbf{R}, c) \geq s_r(\mathbf{R}, c')$ for all candidates c'. The set $r(\mathbf{R})$ contains all winners.

A positional scoring rule r is *two-valued* if there are only two values in each scoring vector $\vec{s}_m$. For two-valued rules, we assume without loss of generality that $\vec{s}_m$ consists only of zeros and ones, and hence is of the form $\vec{s}_m = (1, \dots, 1, 0, \dots, 0)$. We can therefore denote any two-valued rule as k-approval, where $k = k(m)$ may depend on the number m of candidates. For example, $(m-2)$-approval is the same as 2-veto.

For k-approval, we can naturally convert a ranking profile $\mathbf{R} = (R_1, \dots, R_n)$ to an *approval profile* $\mathbf{A} = (A_1, \dots, A_n)$, where each $A_i \subseteq C$ consists of the first k candidates in order R_i. In other words, A_i denotes the candidates voter v_i "approves." The score $s(A_i, c)$ that the voter v_i contributes to the candidate c is one if $c \in A_i$ and zero otherwise. The winners then are the candidates with the maximal score $s(\mathbf{A}, c) = \sum_{i=1}^{n} s(A_i, c)$.

We make some conventional assumptions about the positional scoring rule r. We assume that $\vec{s}_m(j)$ is computable in polynomial time in m, and the scores in each $\vec{s}_m$ are co-prime (i.e., their greatest common divisor is one). A positional scoring rule is called *pure* if every $\vec{s}_{m+1}$ can be obtained from $\vec{s}_m$ by inserting a score value at some position.

2.3 Incomplete Profiles

Throughout this paper, we study problems where voter preferences are incompletely specified and we are interested in "possible" and "necessary" outcomes. Abstractly speaking, an incomplete voting profile is simply a set $\widetilde{\mathcal{R}}$ of ranking profiles. Given $\widetilde{\mathcal{R}}$, a candidate c is called a *possible winner* w.r.t. a voting rule r if c is a winner in at least one profile $\mathbf{R} \in \widetilde{\mathcal{R}}$, that is, $c \in \bigcup_{R \in \widetilde{\mathcal{R}}} r(R)$, and a *necessary winner* w.r.t. r if c is a winner in every profile $\mathbf{R} \in \widetilde{\mathcal{R}}$, that is, $c \in \bigcap_{R \in \widetilde{\mathcal{R}}} r(R)$. It follows from this definition that necessary winners are always possible winners. In contrast to possible winners, necessary winners may not exist.

Incomplete profiles give rise to challenging computational problems since they are represented in a compact manner. For example, in the seminal work of Konczak and Lang [15], an incomplete profile is represented by using a partial order P_i instead of a ranking R_i for every voter v_i. The set of all possible ranking profiles is then defined as the set of profiles that can be obtained by completing every partial order into a total order. In the next section, we discuss another compact representation.

3 The Model of Partial Spatial Voting

We introduce a model of incompleteness for spatial voting. A *partial spatial profile* $\mathbf{P} = (P_1, \dots, P_n)$ consists of a vector of pairs $P_j = \langle (\ell_{j,1}, u_{j,1}), \dots, (\ell_{j,d}, u_{j,d}) \rangle$ for every voter v_j. Each pair $(\ell_{j,i}, u_{j,i})$ represents a lower bound and an upper bound on the position of voter v_j regarding issue i. Note that in this model, the positions of the voters are incompletely specified, but the positions of the candidates are all known precisely. A spatial voting profile $\mathbf{T} = (T_1, \dots, T_n)$ is a *spatial completion* of $\mathbf{P}$ if for every voter v_j and issue $i \in [d]$ we have $T_{j,i} \in [\ell_{j,i}, u_{j,i}]$. We can then compute a ranking profile $\mathbf{R_T}$ as before.

We call a ranking profile $\mathbf{R}$ a *ranking completion* of $\mathbf{P}$ if there exists a spatial completion $\mathbf{T}$ such that $\mathbf{R} = \mathbf{R_T}$. For k-approval, it will be useful to convert the ranking profile to an approval profile, as described in Section 2. We call an approval profile $\mathbf{A}$ an *approval completion* of $\mathbf{P}$ if there exists a spatial profile $\mathbf{T}$ such that $\mathbf{R_T}$ is converted to $\mathbf{A}$.

Given a partial spatial profile $\mathbf{P}$, a candidate c is a *necessary winner* if c is a winner in every ranking completion $\mathbf{R}$ of $\mathbf{P}$, and c is a *possible winner* if there exists a ranking completion $\mathbf{R}$ of $\mathbf{P}$ where c is a winner. For a positional scoring rule r and dimension d, we consider the decision problems where we are given a set C of d-dimensional candidates, a partial profile $\mathbf{P}$ and a candidate $c \in C$, and need to determine whether c is a possible or a necessary winner. We denote these two problems by $\mathsf{PW}\langle d \rangle$ and $\mathsf{NW}\langle d \rangle$, respectively. Note that the dimension is fixed and not part of the input of the problem.

3.1 Partial Order Profiles

The seminal work of Konczak and Lang [15] introduced the model of *partial order profiles* and the problems of possible and necessary winners. In this model, a *partial order profile* $\mathbf{P} = (P_1, \dots, P_n)$ consists of n partial orders (i.e., reflexive, anti-symmetric and transitive relations) on the set C of candidates, where each P_i represents the incomplete preferences of the voter v_i. A *ranking completion* of $\mathbf{P}$ is a ranking profile $\mathbf{R} = (R_1, \dots, R_n)$ where each R_i is a completion (i.e., linear extension) of the partial order P_i.

The possible and necessary winners of a partial order profile $\mathbf{P}$ are defined in the same manner as for spatial order profiles. The decision problems associated with a positional scoring rule r of determining, given a partial order profile $\mathbf{P}$ and a candidate c, whether c is a necessary and a possible winner, are denoted by NW and PW, respectively. A classification of the complexity of these problems has been established in a sequence of publications.

Theorem 1 (Classification Theorem [3, 24, 2])**.** NW *can be solved in polynomial time for every positional scoring rule.* PW *is solvable in polynomial time for plurality and veto; for all other pure scoring rules,* PW *is NP-complete.*

3.2 Relation of Partial Spatial and Partial Order Profiles

We conclude this section with a note on the expressiveness of partial spatial voting in comparison to the model of partial-order profiles. We say that a partial profile $\mathbf{P}$ (in one of the two models) can be *expressed* by the other model if there exists a partial profile $\mathbf{P}'$ in the other model with the same set of ranking completions. Note that in the case of full information, every (complete) profile can be expressed by a spatial profile with $d \leq \min\{m, n\}$ dimensions [5].

For every number d of issues, we can easily come up with partial-order profiles (and even complete ranking profiles) that cannot be expressed as partial spatial profiles, simply by using the property that in spatial voting all voters must respect the positions of the candidates, where in partial orders each voter can have a completely different structure. Note, for example, that if $d = 1$, then all preferences will be single-peaked. Moreover, even for a single voter, there exist partial orders that cannot be expressed as a partial spatial voter, since the two models have different limitations on the number of ranking completions. A single partial order can have up to $m!$ completions, and we later show in Lemma 2 that the number of ranking completions of a partial spatial voter is at most $\mathcal{O}(dm^{2d})$.

On the other hand, consider the case of a single dimension and a set $C = \{c_1, c_2, c_3\}$ of three candidates with the positions $\mathbf{c}_1 = 1, \mathbf{c}_2 = 2$ and $\mathbf{c}_3 = 3$. Consider also a voter v with $P = (1, 3)$. The voter has four ranking completions: (c_1, c_2, c_3), (c_2, c_1, c_3), (c_2, c_3, c_1) and (c_3, c_2, c_1). This incomplete voting profile cannot be expressed with partial orders. Assume to the contrary that there exists a partial order $\succ_v$ with the same set of ranking completions. For every pair $c_i, c_j \in C$, there are ranking completions where c_i is ranked higher than c_j and also rankings where c_j is ranked higher than c_i, hence $c_i \not\succ_v c_j$ and $c_j \not\succ_v c_i$. We get that $\succ_v$ is empty, which results in six ranking completions instead of four. Therefore the partial spatial voter in our example cannot be expressed as a partial order.

We can conclude that complexity results on possible/necessary winners for the partial-order model do not immediately imply results for the partial spatial model, and vice versa, since neither of the two models generalizes the other.

4 Computing Necessary Winners

In this section, we show that the necessary winner problem can be solved in polynomial time for every positional scoring rule and for every fixed number of dimensions. Hence, the tractable cases of NW of partial orders are also tractable for partial spatial profiles.

Theorem 2. *Let* $d \geq 1$ *be fixed.* $\mathsf{NW}\langle d\rangle$ *is solvable in polynomial time for every positional scoring rule.*

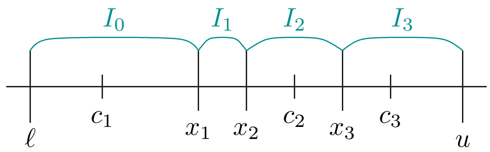


Figure 2: An illustration of the proof of Lemma 1. The three candidates in $C = \{c_1, c_2, c_3\}$ are positioned on a line, and a voter can be positioned at any value in $[\ell, u]$. The values with equal distance between two candidates are x_1 (for c_1 and c_2), x_2 (for c_1 and c_3) and x_3 (for c_2 and c_3). The sequence of intervals is I_0, I_1, I_2, I_3.

The remainder of this section is devoted to proving Theorem 2. To determine whether a candidate c is a necessary winner for a given a partial spatial profile $\mathbf{P}$, we use the same concept from the algorithm for NW given partial orders [24]: c is *not* a necessary winner if and only if there exists another candidate c' and a ranking completion $\mathbf{R}$ where $s(\mathbf{R}, c') > s(\mathbf{R}, c)$. To this end, we iterate over every other candidate c' and compute the maximal score difference $s(\mathbf{R}, c') - s(\mathbf{R}, c)$ among the ranking completions $\mathbf{R}$ of $\mathbf{P}$. Observe that it is sufficient to consider each voter $v_j \in V$ separately and compute the maximal score difference $s(R_j, c') - s(R_j, c)$ among the ranking completions R_j of P_j, since we can sum these values to obtain the maximal value of $s(\mathbf{R}, c') - s(\mathbf{R}, c)$. Then, c is not a necessary winner if and only if the maximal score difference is positive for some candidate c'.

The difference between our algorithm and the one for partial orders is the way we compute the maximal score difference for an individual voter v_j. We show that for a partial spatial voter, we can enumerate all the ranking completions in polynomial time and compute the score difference in each ranking. This is not possible for partial orders since the number of ranking completions in that model can be exponential in m.

Next, we prove that we can indeed enumerate the ranking completions of a single voter in polynomial time. We start with an algorithm for the one-dimensional case and then generalize it to any fixed dimension $d > 1$.

4.1 The Case of a Single Issue

We now assume that the number of issues, d, is 1. In this case, every candidate c_i is associated with a single real value $\mathbf{c}_i$. We assume without loss of generality that the candidates $c_1, \dots, c_m$ are ordered such that $\mathbf{c}_1 < \cdots < \mathbf{c}_m$. A partial profile $\mathbf{P}$ consists of a pair $P_j = (\ell_j, u_j)$ for every voter v_j, and a spatial completion satisfies $T_j \in [\ell_j, u_j]$. Note that this means that in every completion, the preferences of each voter are single-peaked with respect to the candidate order described above.

Lemma 1. *Let C be a set of m 1-dimensional candidates and $P = (\ell, u)$ a 1-dimensional partial spatial vote. The number of ranking completions of P is at most $\binom{m}{2} + 1$. Furthermore, we can enumerate these rankings in polynomial time.*

Proof. Let X be the set of values $x \in [\ell, u]$ such that there exist at least two candidates c, c' for which $|x - \mathbf{c}| = |x - \mathbf{c}'|$. Note that $|X| \leq \binom{m}{2}$ since each pair of candidates introduces at most one value in X. Let $x_1 \leq \cdots \leq x_t$ be the elements of X, and define a sequence $I_0, \dots, I_t$ of intervals such that $I_0 = [\ell, x_1)$, $I_j = [x_j, x_{j+1})$ for every $j \in [t-1]$, and $I_t = [x_t, u]$. The set X and intervals $I_0, \dots, I_t$ are illustrated in Figure 2.

Let $T < T'$ be two spatial completions of P (i.e., values in $[\ell, u]$) that belong to the same interval I_j. Observe that for every pair c and c' of candidates it holds that $|T - \mathbf{c}| > |T - \mathbf{c}'|$ if and only if $|T' - \mathbf{c}| > |T' - \mathbf{c}'|$, since otherwise there exists a value $x \in [T, T']$ such that

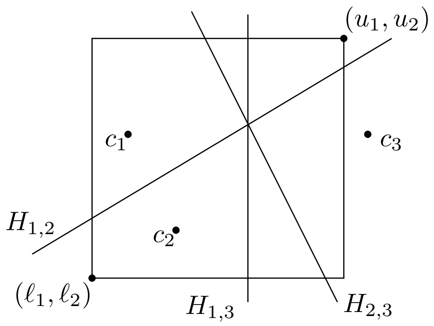


Figure 3: An illustration of the proof of Lemma 2 for $d = 2$ dimensions. Three candidates $C = \{c_1, c_2, c_3\}$ are positioned on a two-dimensional plane. A voter can be positioned at any point in the axis-parallel rectangle that is defined by the interval $[\ell_1, u_1]$ for the horizontal axis and $[\ell_2, u_2]$ for the vertical axis. The line $H_{i,j}$ (a $(d-1)$-dimensional hyperplane) partitions $\mathbb{R}^2$ into 2 regions: the points that are closer to c_i, and the points that are closer to c_j. The top-left region — above $H_{1,2}$ and to the left of $H_{2,3}$ and $H_{1,3}$ — corresponds to the possible positions of the voter where the preference ranking equals $c_1 \succ c_2 \succ c_3$.

$|x - \mathbf{c}| = |x - \mathbf{c}'|$, which implies that the completions belong to different intervals. Therefore the ranking completions R_T and $R_{T'}$ are identical.

We can conclude that the number of ranking completions of P is at most $t+1 \le \binom{m}{2}+1$. To enumerate these rankings, we can select an arbitrary value T in each interval I_j and compute the ranking R_T. □

4.2 The Case of Multiple Issues

We now consider the setting where the number of issues, d, is larger than 1. Recall that every candidate c_i is associated with a d-dimensional real vector $\mathbf{c}_i = \langle c_{i,1}, \ldots, c_{i,d} \rangle \in \mathbb{R}^d$. A partial profile $\mathbf{P}$ consists of a vector of pairs $P_j = \langle (\ell_{j,1}, u_{j,1}), \ldots, (\ell_{j,d}, u_{j,d}) \rangle$ for every voter v_j, and a completion satisfies $T_{j,i} \in [\ell_{j,i}, u_{j,i}]$ for each issue $i \le d$. This means that a voter's possible location is bounded by an axis-parallel d-dimensional rectangle.

As we did for the one-dimensional case, we are again able to bound the number of rankings that a single partial vote can generate — and, more importantly, we show how to efficiently iterate over them. Note that we allow the bounds in this chapter to be exponential in d, as we treat this variable as a constant (rather than a part of the input).

Lemma 2. *Let C be a set of m d-dimensional candidates and $P = \langle (\ell_1, u_1), \ldots, (\ell_d, u_d) \rangle$ a d-dimensional partial spatial vote. The number of ranking completions of P is at most $\mathcal{O}(dm^{2d})$.*

Proof. Every pair of candidates $c, c' \in \mathbb{R}^d$ corresponds to a $(d-1)$-dimensional hyperplane partitioning $\mathbb{R}^d$ into 2 regions or d-faces: the halfspace of points that are closer to c, and the halfspace of points that are closer to c'. Likewise, the set of all (at most) $\binom{m}{2} = m(m-1)/2$ hyperplanes then partitions $\mathbb{R}^d$ into a set of regions, or more formally, of (d-)faces Φ. See the illustration in Figure 3. Note that there is a one-to-one relationship between these faces and the possible rankings of the candidates C: A face $\varphi \in \Phi$ consists exactly of those points in $\mathbb{R}^d$ where the ranking of candidates in C according to distance does not change, and no two of these faces correspond to the same ranking of candidates.

The maximum number of rankings possible is thus equal to the maximum number of d-faces one can partition $\mathbb{R}^d$ into with $m(m-1)/2$ many $(d-1)$-dimensional hyperplanes. It is known that this maximum number is $\sum_{i=0}^{d} \binom{m(m-1)/2}{i}$ (see, e.g., [18, Proposition 6.1.1]). We obtain the asymptotic bound by

$$\sum_{i=0}^{d} \binom{m(m-1)/2}{i} \leq (d+1) \max_{i \leq d} \binom{m(m-1)/2}{i} \leq (d+1)(m(m-1))^d \in \mathcal{O}(dm^{2d}). \quad \square$$

The above result only provides us with an upper bound on the number of rankings. It is not immediately clear how to efficiently iterate over them (or the corresponding d-faces in $\mathbb{R}^d$), as—in contrast to the one-dimensional case—in $\mathbb{R}^d$ we do not have an obvious ordering of the faces that we could iterate over. It is not even clear from the set of hyperplanes itself, which combinations of hyperplanes correspond to one of the faces. Nevertheless, we establish an efficient algorithm to enumerate the rankings.

Lemma 3. *Let C be a set of m d-dimensional candidates and $P = \langle(\ell_1, u_1), \ldots, (\ell_d, u_d)\rangle$ a d-dimensional partial spatial voter. The set of ranking completions of P can be enumerated in polynomial time.*

Proof. The algorithm to enumerate all (at most $\mathcal{O}(dm^{2d})$) different possible rankings uses the idea of considering the hyperplanes corresponding to pairs of points in C again. A pseudocode of the algorithm is given (as Algorithm 1) in the Appendix.

Given the candidates C we can easily compute the corresponding set of at most $m(m-1)/2$ hyperplanes H. We are now interested in constructing a representation of the *arrangement* of these hyperplanes, i.e., of the geometric relation of the (d-)faces spanned by the hyperplanes. A classic approach from computational geometry is to construct an *incidence graph* of the arrangement that consists, basically, of a node for each face of the arrangement, i.e., a node for each point, line(-segment), plane(-segment), etc., where two or more hyperplanes intersect. Further, two nodes are connected by an edge if the corresponding faces are incident, i.e., one is contained in the other. Using an incremental algorithm, where the hyperplanes are introduced one-by-one to the arrangement and the incidence graph is updated accordingly, Edelsbrunner et al. [7] show that the incidence graph of our arrangement can be constructed in $\mathcal{O}(m^{2d})$ time. We denote the incidence graph describing the arrangement of the hyperplanes in H by $G(H)$.

To obtain and iterate over the set of actual possible rankings, we need to find the d-faces of the arrangement intersecting the rectangle corresponding to $\mathbf{P}$. For this, we iterate over those nodes in $G(H)$ that correspond to d-faces of the arrangement. We know from Lemma 2 that there are only polynomially many such nodes. For each such node x we consider all $(d-1)$-faces incident to x to obtain the set of linear inequalities that describe the face corresponding to x. (Note that it is possible to turn a hyperplane into a linear inequality in linear time.) The rectangle corresponding to $\mathbf{P}$ can be expressed via $2d$ linear inequalities itself. The d-face intersects the rectangle if and only if the Linear Feasibility Program (LFP) of the corresponding set of linear inequalities is solvable. For each d-face of the arrangement we therefore solve an LFP with d variables and at most $\frac{m(m-1)}{2} + 2d$ inequalities which can be done in polynomial time [13]. If there is a solution to the LFP we know that $\mathbf{P}$ and the face intersect and a feasible point we obtain from solving the LFP can be used as a witness. Afterwards, by iterating over all witnesses and ordering the candidates by distance from that witness we can iterate over all different possible rankings. $\square$

With this result, we complete the proof of Theorem 2 for higher dimensions.

5 Computing Possible Winners

We now turn to the problem of computing possible winners. Again, we start with the one-dimensional case.

5.1 The Case of a Single Issue

We first assume that $d = 1$ and study the complexity of $\mathsf{PW}\langle 1\rangle$. Recall that every candidate c is associated with a single real value $\mathbf{c}$, that without loss of generality $\mathbf{c}_1 < \cdots < \mathbf{c}_m$, and a partial profile $\mathbf{P}$ consists of a pair $P_j = (\ell_j, u_j)$ for every voter v_j.

For partial orders, by the Classification Theorem (Theorem 1), finding the possible winners is NP-complete for every pure positional scoring rule except for plurality and veto. In spatial voting, we are able to provide efficient algorithms to decide the question for multiple well-studied classes of scoring rules. We begin our investigation with two-valued scoring rules (including plurality and veto), and then prove tractability for other classes of positional scoring rules.

5.1.1 Two-Valued Rules

We first focus on positional scoring rules consisting of two values (zero and one w.l.o.g.). The simplest and most well-known rules in this class are plurality and veto.[2]

We extend the tractability of $\mathsf{PW}\langle 1\rangle$ to any two-valued rule. Recall that we denote a two-valued rule as k-approval for $k = k(m)$. We introduce an alternative definition for partial spatial profiles for k-approval, in the case of a single dimension. Let $\mathbf{P} = (P_1, \dots, P_n)$ be a partial spatial profile where every voter v_j is associated with a pair $P_j = (\ell, u)$. Since we assume that $\mathbf{c}_1 < \cdots < \mathbf{c}_m$, the set of candidates that v_j approves among the completion of P_j is a sequence $(c_{i_j}, c_{i_j+1}, \dots, c_{i_j+t})$ of consecutive candidates. (Note that we can find this sequence for each voter in polynomial time using Lemma 1.) Moreover, in every completion, the candidates that v_j approves are a substring of length k of $(c_{i_j}, c_{i_j+1}, \dots, c_{i_j+t})$.

We can therefore define a partial spatial profile $\mathbf{P} = (P_1, \dots, P_n)$ for k-approval as follows. Each voter v_j is associated with a sequence of at least k consecutive candidates $P_j = (c_\ell, c_{\ell+1}, \dots, c_u)$. In an approval completion $\mathbf{A} = (A_1, \dots, A_n)$ of $\mathbf{P}$, the set A_j of candidates that v_j approves is a substring of length k of P_j. We then use the approval profile $\mathbf{A}$ to compute the scores of the candidates and select the winners (as defined in Section 2.2).

In the next two lemmata, we use this definition of partial votes to show that any partial profile can be converted to another partial profile with some useful properties, and that we can use these properties to solve $\mathsf{PW}\langle 1\rangle$ in polynomial time for two-valued rules.

Lemma 4. *Let $d = 1$ and consider a partial spatial profile $\mathbf{P}$ on a set C of m candidates Furthermore, let r be a two-valued rule. For each candidate $c \in C$, there exists a partial spatial profile $\mathbf{P}'$ with the following properties:*

(i) In $\mathbf{P}'$, each voter either necessarily approves c or never approves c; and

(ii) c is a possible winner in $\mathbf{P}'$ (w.r.t. rule r) if and only if c is a possible winner in $\mathbf{P}$.

Moreover, $\mathbf{P}'$ can be constructed in polynomial time.

The proof of Lemma 4 is given in the Appendix. Next, we show that for k-approval, we can solve the possible winner problem in polynomial time if the partial profile satisfies the properties of $\mathbf{P}'$ in Lemma 4. We use a reduction to scheduling with release times and deadlines, which is defined as follows.

Definition 1 (Non-preemptive multi-machine scheduling with arrival times and deadlines)**.** *We are given a set $M = \{M_1, \dots, M_t\}$ of identical machines and a set $\mathbf{J} = \{J_1, \dots, J_n\}$ of n jobs. Each job J_j has an arrival time a_j, a deadline d_j, and processing time p_j. We*

[2]We later show that for plurality and veto, $\mathsf{PW}\langle d\rangle$ is solvable in polynomial time for every fixed $d \geq 1$. In particular, the tractability of $\mathsf{PW}\langle 1\rangle$ follows both from Theorem 3 (proved in this section) and from Theorem 6 (proved in Section 5.2.

assume that $a_j, d_j, p_j \in \mathbb{N}$. A feasible schedule is a mapping $f\colon \mathbf{J} \to \mathbb{R} \times M$ that maps each $J_j \in \mathbf{J}$ to a pair $f(J_j) = (s_j, h_j)$ such that the following properties hold:

1. *Every job is run between its arrival time and its deadline: $a_j \leq s_j \leq d_j - p_j$ for all $j \in [n]$.*
2. *Each machine can run at most one job at a time: if $h_i = h_j$ for $i, j \in [n]$ with $i \neq j$ and $s_i \leq s_j$ then $s_j \geq s_i + p_i$.*

Since the arrival times, deadlines, and processing times are all integers, we can assume without loss of generality that the starting time of every job in a feasible schedule is also an integer.[3]

We now present the algorithm for possible winner.

Lemma 5. *Let $k = k(m)$. Given a set C of m 1-dimensional candidates, a candidate $c \in C$, and a partial spatial profile $\mathbf{P}$ where every voter either necessarily approves c or never approves c, we can decide whether c is a possible winner in $\mathbf{P}$ w.r.t. k-approval in polynomial time.*

Proof sketch. We show a reduction to multi-machine scheduling where all jobs have the same processing time p. In this case, deciding whether a feasible schedule exists is known to be solvable in polynomial time [21].

Let V_c be the set of voters that necessarily approve c (and the voters of $V \setminus V_c$ never approve c). Note that $s(\mathbf{A}, c) = |V_c|$ in every approval completion $\mathbf{A}$ of $\mathbf{P}$. In the reduction, each voter is a job, and the number of machines is $|V_c|$. Let $v_j \in V$ with $P_j = (c_\ell, \ldots, c_u)$. Define a job J_j with arrival time ℓ and deadline $u + 1$. The processing time of every job is k. In the Appendix, we show that c is a possible winner in $\mathbf{P}$ w.r.t. k-approval if and only if there exists a feasible schedule for all jobs. ☐

Combining Lemma 4 and Lemma 5, we conclude the following.

Theorem 3. $\mathsf{PW}\langle 1\rangle$ *is solvable in polynomial time under every two-valued positional scoring rule.*

5.1.2 Beyond Two-Valued Rules

Next, we discuss rules with more than two values. We start with a family of rules that we refer to as *weighted veto rules.* These rules are of the form $\vec{s}_m = (\alpha, \ldots, \alpha, \beta_1, \ldots, \beta_k)$ for $\alpha > \beta_1 \geq \cdots \geq \beta_k$ and $k < m/2$. The condition $k < m/2$ implies that each voter assigns the highest score α to more than half of the candidates.

Theorem 4. $\mathsf{PW}\langle 1\rangle$ *is solvable in polynomial time under every weighted veto rule.*

Proof. Consider a weighted veto rule with $\vec{s}_m = (\alpha, \ldots, \alpha, \beta_1, \ldots, \beta_k)$. Let C be a set of 1-dimensional candidates and $\mathbf{P}$ a partial profile. We partition the candidates into three sets $C_{\mathsf{lft}} = \{c_1, \ldots, c_k\}$, $C_{\mathsf{rgt}} = \{c_{m-k+1}, \ldots, c_m\}$ and $C_{\mathsf{mid}} = C \setminus (C_{\mathsf{lft}} \cup C_{\mathsf{rgt}})$. Note that $C_{\mathsf{mid}} \neq \emptyset$ since $k < m/2$. Each of the scores $\beta_1, \ldots, \beta_k$ can only be assigned to the candidates of $C_{\mathsf{lft}} \cup C_{\mathsf{rgt}}$, and every voter assigns the score α to the candidates of C_{mid}. Therefore for every ranking completion $\mathbf{R}$ of $\mathbf{P}$, every candidate $c \in C_{\mathsf{mid}}$ receives the maximal possible score $s(\mathbf{R}, c) = n \cdot \alpha$. We can deduce that all candidates of C_{mid} are possible winners. To complete the proof we describe how we can find the possible winners of $C_{\mathsf{lft}} \cup C_{\mathsf{rgt}}$ in polynomial time.

[3] Otherwise, we can iterate over the jobs, sorted from the smallest starting time to the largest. If a job J_i starts at time s_i which is not an integer, we can change the start time to $\lfloor s_i \rfloor$ without changing the feasibility.

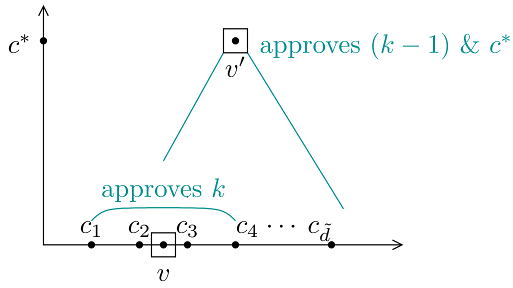


Figure 4: An example of two voters in a completion of the partial profile $\mathbf{P}$ from the proof of Theorem 7. The voter v represents a job of length k, and approves the k candidates closest to it among $c_1, \dots, c_{\tilde{d}}$. The voter v' represents a job of length $k-1$. It approves c^* and the $k-1$ candidates closest to it among $c_1, \dots, c_{\tilde{d}}$.

Let $c \in C_{\mathsf{lft}} \cup C_{\mathsf{rgt}}$. In order to be a winner in a ranking completion $\mathbf{R}$, c must receive the maximal possible score $n \cdot \alpha$, otherwise it is defeated by the candidates of C_{mid}. Hence, c is a possible winner if and only if there exists a completion where every voter assigns the score α to c. We can therefore consider each voter separately, check if it can assign α to c in some completion using Lemma 1, and determine that c is a possible winner if and only if that condition is satisfied for all voters. □

For two positive integers k and t, we denote by $F(k,t)$ the three-valued rule with scoring vector $\vec{s}_m = (2, \dots, 2, 1, \dots, 1, 0, \dots, 0)$ that begins with k occurrences of two and ends with t zeros. For example, the scoring vector for $F(1,1)$ is $\vec{s}_m = (2, 1, \dots, 1, 0)$. In the Appendix, we prove the following.

Theorem 5. $\mathsf{PW}\langle 1 \rangle$ *is solvable in polynomial time under* $F(k,t)$ *whenever* $k > t$*.*

5.2 The Case of Multiple Issues

We return to the setting where $d > 1$. We show that the tractable cases of possible winners for the partial orders model, namely plurality and veto, are also tractable for the spatial model in any fixed dimension d.

Theorem 6. $\mathsf{PW}\langle d \rangle$ *is solvable in polynomial time for plurality and veto.*

Proof. We show a reduction to PW under plurality and veto, respectively, in the model of partial orders, which is known to be solvable in polynomial time [3]. We start with the case of plurality. Let C be a set of candidates and $\mathbf{P} = (P_1, \dots, P_n)$ a partial spatial profile. For every voter v_i, let $C_i \subseteq C$ be the set of candidates that are ranked in the first position in at least one ranking completion of P_i. We can compute C_i in polynomial time using Lemma 3. Define a partial order $P'_i = (C_i \succ (C \setminus C_i))$. It is easy to verify that the ranking completions of P_i and P'_i result in the same possible scores, since only the first candidate in the ranking receives 1 and the others receive 0. Therefore a candidate c is a possible winner for $\mathbf{P}$ if and only if c is a possible winner for $\mathbf{P}' = (P'_1, \dots, P'_n)$.

For veto we use the same idea with a small modification. For every voter v_i, let $C_i \subseteq C$ be the set of candidates that can be ranked last. Define a partial order $P'_i = ((C \setminus C_i) \succ C_i)$. □

For k-approval, where we can find the possible winners in polynomial time for a single dimension (Theorem 3), the problem becomes intractable for $d \geq 2$, and every $k \geq 3$.

Theorem 7. *Let $d \geq 2$ and $k \geq 3$ be fixed.* $\mathsf{PW}\langle d \rangle$ *is NP-complete for k-approval.*

Proof sketch. We focus on $d = 2$, since this is a special case of any higher-dimensional instance. Hence, each candidate c_i is associated with a pair of values $\mathbf{c}_i = \langle c_{i,1}, c_{i,2} \rangle$ and a partial voter consists of two intervals $P_j = \langle (\ell_{j,1}, u_{j,1}), (\ell_{j,2}, u_{j,2}) \rangle$. We show a reduction from scheduling with arrival times and deadlines (from Definition 1) where we have a single machine and every processing time satisfies $p_j \in \{k, k-1\}$. Deciding whether a feasible schedule exists is strongly NP-complete for every pair of job lengths $k, k-1$ where $k > 2$ [8]. Since it is strongly NP-complete, we can assume that the maximal deadline $d_{\max}$ is polynomial in the number of jobs n. We also assume that the minimal arrival time is 1.

Let $\mathbf{J} = \{J_1, \dots J_n\}$ be the set of jobs, each job J_j with arrival time a_j and deadline d_j. We partition the jobs into two sets $\mathbf{J}_{k-1}, \mathbf{J}_k$ depending on their processing time. The reduction is as follows. Let $\tilde{d}$ be the smallest multiplication of k that is greater or equal to $d_{\max} - 1$, the candidates are $C = \left\{c^*, c_1, \dots, c_{\tilde{d}}\right\}$. The position of c^* is $\mathbf{c}^* = \langle 0, 3\tilde{d} \rangle$ and the position of every other candidate c_i is $\mathbf{c}_i = \langle i + 0.5, 0 \rangle$. For the partial voters that we define, the interval for the first position $(\ell_{j,1}, u_{j,1})$ always satisfies $0 \le \ell_{j,1} \le u_{j,1} \le \tilde{d}$, and the position on the second issue is always one of two values, either 0 or $3\tilde{d}$.

Observe that for a partial voter v with $P = \langle (\ell, u), 0 \rangle$ such that $0 \le \ell \le u \le \tilde{d}$, the set of candidates that v approves among the completions of P is a sequence of consecutive candidates $(c_i, c_{i+1}, \dots, c_{i+t})$. Moreover, in every completion, the candidates that v approves are a substring of length k of that sequence. If we have $P = \langle (\ell, u), 3\tilde{d} \rangle$ then v necessarily approves c^*. Besides c^*, the set of candidates that v approves among the completions of P is again a sequence of consecutive candidates, and in each completion, v approves a substring of length $k - 1$ of that sequence. We can therefore use an alternative definition for partial voters as in Section 5.1.1. For each voter we specify the value on the second issue, and a sequence of candidates that the voter approves among the completions (not including c^*, since each voter either necessarily approves it or never approves it). Hence we denote $P = \langle (c_i, \dots, c_{i+t}), y \rangle$ where $(c_i, \dots, c_{i+t})$ is a sequence of consecutive candidates and $y \in \{0, 3\tilde{d}\}$. An approval completion is defined as before (for each voter, we specify the candidates that it approves).

The partial profile $\mathbf{P} = \mathbf{P}^1 \circ \mathbf{P}^2 \circ \mathbf{P}^3$, illustrated in Figure 4, consists of three parts that we describe next. In the first part, $\mathbf{P}^1$, for every job $J_j \in \mathbf{J}_k$ we introduce a voter v_j with $P_j = \langle (c_{a_j}, \dots, c_{d_j - 1}), 0 \rangle$. In the second part, $\mathbf{P}^2$, for every job $J_j \in \mathbf{J}_{k-1}$ we introduce a voter v_j with $P_j = \langle (c_{a_j}, \dots, c_{d_j - 1}), 3\tilde{d} \rangle$. The third part $\mathbf{P}^3$ consists of $(|\mathbf{J}_{k-1}| - 1) \cdot \tilde{d}/k$ voters without uncertainty such that every candidate among $c_1, \dots, c_{\tilde{d}}$ is approved by exactly $|\mathbf{J}_{k-1}| - 1$ voters. Specifically, $|\mathbf{J}_{k-1}| - 1$ voters approve $c_1, \dots, c_k$, then $|\mathbf{J}_{k-1}| - 1$ voters approve $c_{k+1}, \dots, c_{2k}$ and so on. In the Appendix, we show that c^* is a possible winner of $\mathbf{P}$ if and only if there exists a feasible schedule. $\square$

6 Conclusions

We introduced the framework of partial spatial voting, where candidates and voters are positioned in a geometrical space, but voters can have intervals of possible values instead of single values in each dimension (that corresponds to an issue). For positional scoring rules, we recovered the tractable cases of necessary and possible winners in the model of partial orders, for every fixed number of issues. In particular, we showed that the possible winners can be found in polynomial time for the plurality and veto rules, and that the necessary winners can be found in polynomial time for every positional scoring rule. We also identified cases where the possible-winner problem is hard for partial orders but not for partial spatial voting. Specifically, we showed it for the two-valued rules other than plurality and veto, such as k-approval and k-veto for $k > 1$. We also showed that the possible-winner problem may become intractable when we increase the number of issues to a higher fixed number.

Acknowledgments. The work of Markus Brill and Jonas Israel was supported by the Deutsche Forschungsgemeinschaft under grant BR 4744/2-1. The work of Aviram Imber and Benny Kimelfeld was supported by the US-Israel Binational Science Foundation (BSF) under Grant 2017753.

References

[1] K. J. Arrow. *Social Choice and Individual Values.* New Haven: Cowles Foundation, 1st edition, 1951. 2nd edition 1963.

[2] Dorothea Baumeister and Jörg Rothe. Taking the final step to a full dichotomy of the possible winner problem in pure scoring rules. *Inf. Process. Lett.*, 112(5):186–190, 2012.

[3] Nadja Betzler and Britta Dorn. Towards a dichotomy for the possible winner problem in elections based on scoring rules. *J. Comput. Syst. Sci.*, 76(8):812–836, 2010.

[4] D. Black. On the rationale of group decision-making. *Journal of Political Economy*, 56 (1):23–34, 1948.

[5] Anna Bogomolnaia and Jean-François Laslier. Euclidean preferences. *Journal of Mathematical Economics*, 43(2):87–98, 2007.

[6] Jean-Paul Doignon and Jean-Claude Falmagne. A polynomial time algorithm for unidimensional unfolding representations. *Journal of Algorithms*, 16(2):218–233, 1994.

[7] Herbert Edelsbrunner, Joseph O'Rourke, and Raimund Seidel. Constructing arrangements of lines and hyperplanes with applications. *SIAM Journal on Computing*, 15(2): 341–363, 1986.

[8] Jan Elffers and Mathijs de Weerdt. Scheduling with two non-unit task lengths is NP-complete. *CoRR*, abs/1412.3095, 2014.

[9] Edith Elkind and Piotr Faliszewski. Recognizing 1-Euclidean preferences: An alternative approach. In *Proceedings of the 7th International Symposium on Algorithmic Game Theory (SAGT)*, pages 146–157. Springer, 2014.

[10] Edith Elkind, Martin Lackner, and Dominik Peters. Preference restrictions in computational social choice: A survey. *arXiv preprint arXiv:2205.09092*, 2022.

[11] James M Enelow and Melvin J Hinich. *The spatial theory of voting: An introduction.* Cambridge University Press, 1984.

[12] Andrew Estornell, Sanmay Das, Edith Elkind, and Yevgeniy Vorobeychik. Election control by manipulating issue significance. In *Proceedings of Machine Learning Research*, pages 340–349. AUAI Press, 2020.

[13] L. Khachiyan. A polynomial algorithm in linear programming. *Soviet Mathematics Doklady*, 20:191–194, 1979.

[14] V. Knoblauch. Recognizing one-dimensional Euclidean preference profiles. *Journal of Mathematical Economics*, 46(1):1–5, 2010.

[15] Kathrin Konczak and Jérôme Lang. Voting procedures with incomplete preferences. In *Proceedings of the Multidisciplinary IJCAI-05 Workshop on Advances in Preference Handling*, 2005.

[16] Jérôme Lang. Collective decision making under incomplete knowledge: Possible and necessary solutions. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI) Survey Track*, pages 4885–4891. ijcai.org, 2020.

[17] Jasper Lu, David Kai Zhang, Zinovi Rabinovich, Svetlana Obraztsova, and Yevgeniy Vorobeychik. Manipulating elections by selecting issues. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 529–537. IFAAMA, 2019.

[18] Jirí Matousek. *Lectures on Discrete Geometry.* Springer Science & Business Media, 2013.

[19] N. R. Miller. *Committees, Agendas and Voting.* Harwood Academic, 1995.

[20] Dominik Peters. Recognising multidimensional Euclidean preferences. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI).* AAAI press, 2017.

[21] Nodari Vakhania. Branch less, cut more and minimize the number of late equal-length jobs on identical machines. *Theoretical Computer Science*, 465:49–60, 2012.

[22] Toby Walsh. Uncertainty in preference elicitation and aggregation. In *Proccedings of the 21st AAAI Conference on Artifical Intelligence (AAAI)*, pages 3–8. AAAI Press, 2007.

[23] Junlin Wu, Andrew Estornell, Lecheng Kong, and Yevgeniy Vorobeychik. Manipulating elections by changing voter perceptions. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 557–563. ijcai.org, 2022.

[24] Lirong Xia and Vincent Conitzer. Determining possible and necessary winners given partial orders. *J. Artif. Intell. Res.*, 41:25–67, 2011.

Aviram Imber
Technion – Israel Institute of Technology
Haifa, Israel
Email: aviram.imber@cs.technion.ac.il

Jonas Israel
Research Group Efficient Algorithms, Technische Universität Berlin
Berlin, Germany
Email: j.israel@tu-berlin.de

Markus Brill
Department of Computer Science, University of Warwick
Coventry, UK
Email: markus.brill@warwick.ac.uk

Hadas Shachnai
Technion – Israel Institute of Technology
Haifa, Israel
Email: hadas@cs.technion.ac.il

Benny Kimelfeld
Technion – Israel Institute of Technology
Haifa, Israel
Email: bennyk@technion.ac.il

A Proofs for Section 4 (Computing Necessary Winners)

Algorithm 1: Enumerate Rankings

Input : m candidates C as points in $\mathbb{R}^d$ and the rectangle corresponding to $\mathbf{P}$ (via $2d$ linear inequalities)

Output: set of possible rankings of C according to possible positions $\mathbf{P}$

$H \leftarrow \emptyset$ `// set of hyperplanes`
$\Phi \leftarrow \emptyset$ `// set of` d`-faces`
$R \leftarrow \emptyset$ `// set of possible rankings`
foreach *distinct pair* $c, c' \in C$ **do**
  $h \leftarrow$ hyperplane corresponding to "middle" of c and c'
  $H \leftarrow H \cup \{h\}$
$G \leftarrow$ incidence graph of H `// can be obtained as in [7]`
foreach *node* x *of* G *corresp. to a* d*-face* **do**
  $\varphi \leftarrow \emptyset$ `// linear ineq.s bounding` d`-face` x
  foreach *node* y *in* G *corresp. to a* $(d-1)$*-face incident to* x **do**
    $L \leftarrow$ linear inequalities corresponding to y
    $\varphi \leftarrow \varphi \cup L$
  $\Phi \leftarrow \Phi \cup \varphi$
foreach $\varphi \in \Phi$ **do**
  if $\varphi \cap P$ **then**
    $w_\varphi \leftarrow$ point in intersection of φ and $\mathbf{P}$
    $r \leftarrow C$, sorted by distance to w_φ
    $R \leftarrow R \cup \{r\}$
return R

B Proofs for Section 5 (Computing Possible Winners)

Lemma 4. *Let $d = 1$ and consider a partial spatial profile $\mathbf{P}$ on a set C of m candidates Furthermore, let r be a two-valued rule. For each candidate $c \in C$, there exists a partial spatial profile $\mathbf{P}'$ with the following properties:*

(i) In $\mathbf{P}'$, each voter either necessarily approves c or never approves c; and

(ii) c is a possible winner in $\mathbf{P}'$ (w.r.t. rule r) if and only if c is a possible winner in $\mathbf{P}$.

Moreover, $\mathbf{P}'$ can be constructed in polynomial time.

Proof. Let $k = k(m)$ the value for which the two-valued rule r corresponds to k-approval. Given $\mathbf{P} = (P_1, \ldots, P_n)$, let $V_c = \{v_j \in V : c \in P_j\}$ denote the voters that approve c in at least one completion of $\mathbf{P}$. We construct $\mathbf{P}' = (P'_1, \ldots, P'_n)$ as follows. Let v_j be a voter with $P_j = (c_\ell, c_{\ell+1}, \ldots, c_u)$. If $v \notin V_c$ then v never approves c, and we set $P'_j = P_j$. Otherwise, there exists $\ell \leq i \leq u$ for which $c_i = c$. Define $\ell' = \max\{\ell, i-k+1\}$ and $u' = \min\{u, i+k-1\}$. Then, $P'_j = (c_{\ell'}, c_{\ell'+1}, \ldots, c_{u'})$. Observe the following:

- $\ell' \geq \ell$ and $u' \leq u$, hence P'_j is a substring of P_j.
- P'_j consists of at least k candidates.

- Every substring of P_j of length k contains c. I.e., c is necessarily approved by v_j in $\mathbf{P}'$.

We show that c is a possible winner in $\mathbf{P}$ w.r.t. k-approval if and only if c is possible winner in $\mathbf{P}'$. One direction is trivial since every completion of $\mathbf{P}'$ is also a completion of $\mathbf{P}$. For the other direction, assume that there exists an approval completion $\mathbf{A} = (A_1, \dots, A_n)$ of $\mathbf{P}$ where c is a winner. We construct a completion $\mathbf{A}' = (A'_1, \dots, A'_n)$ of $\mathbf{P}'$ as follows. Let v_j be a voter. If $v_j \notin V_c$, or $v_j \in V_c$ and $c \in A_j$, then $A'_j = A_j$. Otherwise, $v_j \in V_c$ and $c \notin A_j$, i.e., V does not approve c in A_j. Let A'_j be an arbitrary completion of P'_j. Note that $c \in A'_j$ by the construction of P'_j. We get that $s(A'_j, c) = s(A_j, c) + 1$, and for every other candidate c', $s(A'_j, c') \leq s(A_j, c') + 1$, hence c remains a winner when we change A_j to A'_j. Overall, c is a winner of $\mathbf{A}'$. ☐

Lemma 5. *Let $k = k(m)$. Given a set C of m 1-dimensional candidates, a candidate $c \in C$, and a partial spatial profile $\mathbf{P}$ where every voter either necessarily approves c or never approves c, we can decide whether c is a possible winner in $\mathbf{P}$ w.r.t. k-approval in polynomial time.*

Proof. Following the proof sketch, we show that c is a possible winner in $\mathbf{P}$ w.r.t. k-approval if and only if there exists a feasible schedule of all the jobs. Assume that there exists an approval completion $\mathbf{A} = (A_1, \dots, A_n)$ of $\mathbf{P}$ where c is a winner. Note that $s(\mathbf{A}, c) = |V_c|$ and $s(\mathbf{A}, c') \leq |V_c|$ for every other candidate $c' \in C$. We define a schedule as follows. For a voter v_j, let $A_j = \left\{c_{i_j}, c_{i_j+1} \dots, c_{i_j+k-1}\right\}$ be the sequence of k consecutive candidates that v_j approves in $\mathbf{A}$. We schedule the job J_j from time i_j to time $i_j + k$. We show that this schedule is feasible for $|V_c|$ machines. By the definition of a completion, no job is started before its arrival and each job is completed by its deadline. Observe that for every $i \in [m]$ and $j \in [n]$, job J_j is scheduled to run at time $[i, i+1]$ if and only if v_j approves c_i in $\mathbf{A}$. Since $s(\mathbf{A}, c') \leq |V_c|$ for every candidate $c' \in C$, we can deduce that any time at most $|V_c|$ jobs are scheduled, hence the schedule is feasible.

Next, assume that there exists a feasible schedule (and recall our assumption that the starting times are integers). We construct a completion $\mathbf{A} = (A_1, \dots, A_n)$ of $\mathbf{P}$ as follows. For every voter v_j, let $[i_j, i_j + k]$ be the scheduled execution time of J_j in the schedule. Define $A_j = \left\{c_{i_j}, c_{i_j+1} \dots, c_{i_j+k-1}\right\}$. Note that by the definition of the arrival time and the deadline, A_j is a substring of P_j of length k, hence each A_j is indeed a completion of P_j, and $\mathbf{A}$ is a completion of $\mathbf{P}$. We show that c is a winner of $\mathbf{A}$.

First, $s(\mathbf{A}, c) = |V_c|$ since $\mathbf{A}$ is a completion of $\mathbf{P}$. Second, observe that for every $i \in [m]$ and $j \in [n]$, the voter v_j approves c_i in $\mathbf{A}$ if and only if the job J_j is scheduled to run at time $[i, i+1]$. Since there are only $|V_c|$ machines, at any given time the number of jobs that are scheduled to run is at most $|V_c|$, which implies that $s(\mathbf{A}, c') \leq |V_c|$ for every candidate $c' \in C$. We deduce that c is a winner in $\mathbf{A}$. ☐

Theorem 5. $\mathsf{PW}\langle 1 \rangle$ *is solvable in polynomial time under $F(k, t)$ whenever $k > t$.*

Proof. Let C be a set of 1-dimensional candidates and $\mathbf{P}$ a partial profile. We denote the rule $F(k, t)$ by r and k-approval by r'. Recall that for a ranking completion $\mathbf{R}$, we denote the score of c in $\mathbf{R}$ w.r.t. a voting rule r by $s_r(\mathbf{R}, c)$. As in the proof of Theorem 4, we partition the candidates to three sets $C_{\mathsf{lft}} = \{c_1, \dots, c_t\}$, $C_{\mathsf{rgt}} = \{c_{m-t+1}, \dots, c_m\}$ and $C_{\mathsf{mid}} = C \setminus (C_{\mathsf{lft}} \cup C_{\mathsf{rgt}})$.

We start by showing a connection between the scores of candidates in $C_{\mathsf{lft}}, C_{\mathsf{rgt}}$ and the scores of two specific candidates $c_{t+1}, c_{m-t} \in C_{\mathsf{mid}}$. Let $\mathbf{R} = (R_1, \dots, R_n)$ be a ranking completion of $\mathbf{P}$, let v_j be a voter, and let $c_{\mathsf{lft}} \in C_{\mathsf{lft}}$. Note that v_j can only assign the score 0 to the candidates of $C_{\mathsf{lft}} \cup C_{\mathsf{rgt}}$, since it assigns 0 to the t farthest candidates from its position, and $C_{\mathsf{lft}} \cup R$ are the first and last t candidates on the line. Hence $s_r(R_j, c) \geq 1$ for every candidate $c \in C_{\mathsf{mid}}$, and in particular $s_r(R_j, c_{t+1}) \geq 1$.

We have three options for the score that v_j assigns to c_{lft}. If the score is 0 then $s_r(R_j, c_{\mathsf{lft}}) \leq s_r(R_j, c_{t+1}) + 1$, and if it is 1 then $s_r(R_j, c_{\mathsf{lft}}) \leq s_r(R_j, c_{t+1})$. If $s_r(R_j, c_{\mathsf{lft}}) = 2$ then we also have $s_r(R_j, c_{t+1}) = 2$ since v_j assigns 2 to $k > t$ candidates, c_{lft} is one of the t left-most candidates, and c_{t+1} is the $(t+1)$th candidate from the left. By summing the scores from all voters, we get the following:

$$s_r(\mathbf{R}, c_{\mathsf{lft}}) \leq s(\mathbf{R}, c_{t+1}) - B(\mathbf{R}, c_{\mathsf{lft}}) \tag{1}$$

where $B(\mathbf{R}, c)$ is the number of voters that assign 0 to c in $\mathbf{R}$. Similarly, for every $c_{\mathsf{rgt}} \in C_{\mathsf{rgt}}$ we have

$$s_r(\mathbf{R}, c_{\mathsf{rgt}}) \leq s(\mathbf{R}, c_{m-t}) - B(\mathbf{R}, c_{\mathsf{rgt}}) \tag{2}$$

We now show a connection between the scores of candidates in a completion w.r.t. r and r'. Let v_j be a voter and let $c_{\mathsf{mid}} \in C_{\mathsf{mid}}$. Recall that $s_r(R_j, c_{\mathsf{mid}}) \geq 1$. If c_{mid} is among the top k candidates in the ranking of v_j then $s_r(R_j, c_{\mathsf{mid}}) = 2$ and $s_{r'}(R_j, c_{\mathsf{mid}}) = 1$. Otherwise, c_{mid} is not among the top k candidates in the ranking of v_j, which implies $s_r(R_j, c_{\mathsf{mid}}) = 1$ and $s_{r'}(R_j, c_{\mathsf{mid}}) = 0$. In both cases we get $s_r(R_j, c_{\mathsf{mid}}) = s_{r'}(R_j, c_{\mathsf{mid}}) + 1$, and overall

$$s_r(\mathbf{R}, c_{\mathsf{mid}}) = s_{r'}(\mathbf{R}, c_{\mathsf{mid}}) + n \tag{3}$$

For a candidate $c_{\mathsf{lft}} \in C_{\mathsf{lft}}$ we have the same relation between the scores under r and r', unless $s_r(R_j, c_{\mathsf{lft}}) = 0$ which implies $s_{r'}(R_j, c_{\mathsf{lft}}) = s_r(R_j, c_{\mathsf{lft}}) = 0$. We can apply the same argument for every $c_{\mathsf{rgt}} \in C_{\mathsf{rgt}}$, and obtain the following.

$$s_r(\mathbf{R}, c_{\mathsf{lft}}) = s_{r'}(\mathbf{R}, c_{\mathsf{lft}}) + n - B(\mathbf{R}, c_{\mathsf{lft}}) \tag{4}$$

$$s_r(\mathbf{R}, c_{\mathsf{rgt}}) = s_{r'}(\mathbf{R}, c_{\mathsf{rgt}}) + n - B(\mathbf{R}, c_{\mathsf{rgt}}) \tag{5}$$

We now present the algorithm to determine whether a candidate c is a winner. We use a different procedure for each set of candidates $C_{\mathsf{lft}}, C_{\mathsf{mid}}, C_{\mathsf{rgt}}$. Let $c \in C_{\mathsf{mid}}$. We show that for every ranking completion $\mathbf{R}$ of $\mathbf{P}$, c is a winner of $\mathbf{R}$ w.r.t. r if and only if c is a winner of $\mathbf{R}$ w.r.t. r'. Finding the possible winners under r' is covered by Theorem 3, hence we get that we can decide whether c is a possible winner w.r.t. r in polynomial time.

Assume that c is a winner of $\mathbf{R}$ w.r.t. r, i.e., $s_r(\mathbf{R}, c) \geq s_r(\mathbf{R}, c')$ for every other candidate c'. Let $c_{\mathsf{mid}} \in C_{\mathsf{mid}}$, by Equation (3) we get $s_{r'}(\mathbf{R}, c) \geq s_{r'}(\mathbf{R}, c_{\mathsf{mid}})$. For every $c_{\mathsf{lft}} \in C_{\mathsf{lft}}$, we apply Equations 1 and 4.

$$s_{r'}(\mathbf{R}, c) = s_r(\mathbf{R}, c) - n \geq s_r(\mathbf{R}, c_{t+1}) - n \geq (s_r(\mathbf{R}, c_{\mathsf{lft}})) + B(\mathbf{R}, c_{\mathsf{lft}})) - n = s_{r'}(\mathbf{R}, c_{t+1})$$

For $c_{\mathsf{rgt}} \in C_{\mathsf{rgt}}$ we apply Equations 2 and 5 in the same manner. We can deduce that c is a winner of $\mathbf{R}$ w.r.t. r'.

Now, assume that c is a winner of $\mathbf{R}$ w.r.t. r'. For every $c_{\mathsf{mid}} \in C_{\mathsf{mid}}$ we can use Equation (3) again to show that $s_r(\mathbf{R}, c) \geq s_r(\mathbf{R}, c_{\mathsf{mid}})$. For $c_{\mathsf{lft}} \in C_{\mathsf{lft}}$ we use Equations 3 and 4.

$$s_r(\mathbf{R}, c) = s_{r'}(\mathbf{R}, c) + n \geq s_{r'}(\mathbf{R}, c_{\mathsf{lft}}) + n \geq s_{r'}(\mathbf{R}, c_{\mathsf{lft}}) + n - B(\mathbf{R}, c_{\mathsf{lft}}) = s_r(\mathbf{R}, c_{\mathsf{lft}})$$

For $c_{\mathsf{rgt}} \in C_{\mathsf{rgt}}$ we apply Equations 3 and 5 in the same manner. We can deduce that c is a winner of $\mathbf{R}$ w.r.t. r. This completes the proof for the algorithm of $\mathsf{PW}\langle 1 \rangle$ in the case that $c \in C_{\mathsf{mid}}$.

We now show an algorithm for $c \in C_{\mathsf{lft}}$, and the case of $c \in C_{\mathsf{rgt}}$ is similar. In a completion $\mathbf{R}$, if $B(\mathbf{R}, c) > 0$ then c is not a winner of $\mathbf{R}$ since $s_r(\mathbf{R}, c) < s_r(\mathbf{R}, c_{t+1})$ by Equation (1).

We define another partial profile $\mathbf{P}'$ where voters never assign 0 to c and get that c is a possible winner of $\mathbf{P}$ if and only if it is a possible winner of $\mathbf{P}'$. Note that $\mathbf{P}'$ can be easily constructed by inspecting the ranking completions of each voter v_j and modifying the values of $P_j = (\ell_j, u_j)$ accordingly.

In every completion $\mathbf{R}$ of $\mathbf{P}'$ we have $s_r(\mathbf{R}, c) = s_{r'}(\mathbf{R}, c) + n$, since we can use Equation (4) with $B(\mathbf{R}, c) = 0$. By the same arguments that we had for the case of $c \in C_{\mathsf{mid}}$, we can show that c is a possible winner of $\mathbf{P}'$ w.r.t. r if and only if it is a possible winner of $\mathbf{P}'$ w.r.t. r'. □

Theorem 7. *Let $d \geq 2$ and $k \geq 3$ be fixed.* $\mathsf{PW}\langle d\rangle$ *is NP-complete for k-approval.*

Proof. Following the proof sketch, we start by analyzing the possible scores in different completions. Let $\mathbf{A} = \mathbf{A}^1 \circ \mathbf{A}^2 \circ \mathbf{A}^3$ be an approval completion of $\mathbf{P}$, we start with some observations regarding the profile. For every voter v_j in $\mathbf{A}^1$, as we stated in the discussion on the alternative model of partial votes, c^* is never approved by v_j, hence $s(\mathbf{A}^1, c^*) = 0$. In contrast, every voter v_j in $\mathbf{A}^2$ approves c^*, which implies $s(\mathbf{A}^2, c^*) = |\mathbf{J}_{k-1}|$. For the third part $\mathbf{A}^3$ we get $s(\mathbf{A}^3, c^*) = 0$ and $s(\mathbf{A}^3, c_i) = |\mathbf{J}_{k-1}| - 1$ for every candidate c_i. Overall, the score of c^* is $s(\mathbf{A}, c^*) = |\mathbf{J}_{k-1}|$.

We now show that c^* is a possible winner of $\mathbf{P}$ if and only if there exists a feasible schedule. Let $\mathbf{A}$ be an approval completion of $\mathbf{P}$ where c^* a winner. By our analysis of the scores in the possible completion, for every candidate c_i we have $s(\mathbf{A}, c_i) \leq s(\mathbf{A}, c^*) = |\mathbf{J}_{k-1}|$. Since $s(\mathbf{A}^3, c_i) = |\mathbf{J}_{k-1}| - 1$ we get $s(\mathbf{A}^1 \circ \mathbf{A}^2, c_i) \leq 1$, i.e., at most one voter from $\mathbf{A}^1 \circ \mathbf{A}^2$ approves c_i. We construct a schedule as follows. For every job $J_j \in \mathbf{J}_k$, let $(c_{i_j}, c_{i_j+1}, \ldots, c_{i_j+k-1})$ be the k candidates that v_j approves in $A_j \in \mathbf{A}^1$. We schedule J_j to start at time i_j. For every job $J_j \in \mathbf{J}_{k-1}$, let $(c_{i_j}, c_{i_j+1}, \ldots, c_{i_j+k-2})$ be the $k-1$ candidates that v_j approves in $A_j \in \mathbf{A}^2$ other than c^*. We again schedule J_j to start at time i_j.

We show that this is a feasible schedule. By the definition of the partial votes, every job is processed between its arrival time and its deadline. Let $c_i \in C$. Observe that a job J_j is scheduled to run at time i if and only if the voter v_j approves c_i in $\mathbf{A}^1 \circ \mathbf{A}^2$. Since At most one voter from $\mathbf{A}^1 \circ \mathbf{A}^2$ approves c_i, we can deduce that at most one job is scheduled to run at time i, therefore we never schedule two jobs at the same time and the schedule is feasible.

We now prove the other direction. Assume there exists a feasible schedule (recall we can assume the starting times are all integers). We construct a completion $\mathbf{A} = \mathbf{A}^1 \circ \mathbf{A}^2 \circ \mathbf{A}^3$ of $\mathbf{P}$. For every job $J \in \mathbf{J}_k$, let i_j be the scheduled starting time, we define $A_j = \{c_{i_j}, c_{i_j+1}, \ldots, c_{i_j+k-1}\}$. For every job $J \in \mathbf{J}_{k-1}$, let i_j be the scheduled starting time, we define $A_j = \{c^*, c_{i_j}, c_{i_j+1}, \ldots, c_{i_j+k-2}\}$. Note that by the definition of the arrival times, deadlines, and the two types of jobs, each A_j is a valid completion of P_j, and $\mathbf{A}$ is a completion of $\mathbf{P}$. We show that c is a winner of $\mathbf{A}$.

By our analysis regarding the scores in different completions, it is sufficient to show that $s(\mathbf{A}^1 \circ \mathbf{A}^2, c_i) \leq 1$ for every candidate c_i. For every candidate c_i and voter v_j, v_j approves c_i in $\mathbf{A}^1 \circ \mathbf{A}^2$ if and only if the job J_j is scheduled to run at time i. Since at most one job is scheduled to run at any time, we get that at most one voter of $\mathbf{A}^1 \circ \mathbf{A}^2$ approves c_i. This completes the proof. □