The Complexity of Recognizing Incomplete Single-Crossing Preferences

Edith Elkind[a], Piotr Faliszewski[b], Martin Lackner[a], Svetlana Obraztsova[c]

[a] *University of Oxford, Oxford, United Kingdom*
[b] *AGH University of Science and Technology, Kraków, Poland*
[c] *Hebrew University of Jerusalem, Jerusalem, Israel*

Abstract

We study the complexity of deciding if a given profile of incomplete votes (i.e., a profile of partial orders over a given set of alternatives) can be extended to a single-crossing profile of complete votes (total orders). This problem models settings where we have partial knowledge regarding voters' preferences and we would like to understand whether the given preference profile may be single-crossing. We show that this problem admits a polynomial-time algorithm when the order of votes is fixed and the input profile consists of weak orders, but becomes NP-complete if we are allowed to permute the votes. Moreover, we identify a number of practical special cases of the latter problem that admit polynomial-time algorithms.

Keywords: Computational social choice, preference domains, recognition algorithms, incomplete preferences

---

1. Introduction

An important job for a designer of a multi-agent system is identifying a good method of aggregating the agents' preferences. It is well-known that this is not an easy task, at least if agents' preferences can be arbitrary total orders over the available alternatives: every preference aggregation mechanism for this setting exhibits undesirable behavior on some inputs [1]. However, the designer's task becomes much easier when agents' preferences possess additional structure.

For instance, the well-known class of *single-peaked preferences* [8] admits a voting rule that always selects a Condorcet winner (an alternative that is preferred to every other alternative by a majority of voters) and is strategyproof [35]. Moreover, single-peaked preferences admit efficient algorithms for problems that are more complex than selecting a single winner and that are known to be hard for general preferences, such as choosing a good ranking of the alternatives [10] or a representative committee [7]. A more detailed discussion of algorithmic advances in detecting and exploiting structured preferences (such as single-peaked preferences) can be found in a recent survey by Elkind et al. [23].

In this paper, we focus on another restricted preference domain, namely, that of *single-crossing preferences*. A preference profile is single-crossing with respect to a fixed ordering of voters if for every pair of alternatives a, b it holds that all voters who prefer a to b precede all voters who prefer b to a or vice versa. A profile is single-crossing if the votes can be permuted so as to achieve the single-crossing property. Single-crossing preferences, originally introduced by Mirrlees [34] and Roberts [39], arise in situations where voters and candidates are spread over a spectrum of opinions—say, from extreme left-wing ones to extreme right-wing ones—and left-leaning voters prefer left-leaning candidates to right-leaning ones, and the other way round for the right-leaning voters. While this domain is perhaps not as well-known as that of single-peaked preferences, it has many of the same desirable properties: for instance, under single-crossing preferences the majority relation is transitive [34], and single-crossing preferences admit efficient algorithms for several voting problems that are hard for the general domain [16, 41, 32].

However, in practice we rarely have access to voters' full preferences: voters are far more likely to only report some part of their preference order, e.g., rank a few top alternatives or report a small number of pairwise comparisons. Indeed, in an overwhelming majority of data sets in `PrefLib` [33] preference profiles contain partial orders. This phenomenon is recognized by computational social choice researchers, who showed that many of the positive results that are known to hold for complete preference profiles can be extended to partial preference profiles [5, 36]. It also motivated research on the *possible/necessary winner problem* [29, 6, 42, 4], where we ask whether a given candidate wins in some/all extensions of a given profile of partial votes to a profile of full votes, under a particular voting rule. In a similar vein, we can ask if a profile of partial votes can be extended so that it enjoys a particular structural property, such as being single-peaked/single-crossing, and, if the answer is positive, whether we can identify an ordering of candidates/voters witnessing this. Answering this question would tell us whether voters' preferences may be essentially one-dimensional in nature; if this answer is positive, we may be able to make a reasonably good decision quickly and without eliciting full preferences.

For incomplete single-peaked preferences, the complexity of this problem has been investigated by Lackner [30], who proved that it is NP-complete when the input may consist of arbitrary partial orders, and more recently by Fitzsimmons [27], who showed it to be polynomial-time solvable for weak orders. The goal of our paper is to initiate the complexity-theoretic investigation of this problem for incomplete single-crossing preferences.

Our Contribution. We consider the complexity of deciding whether a given profile of partial orders can be extended to a profile of total orders that is single-crossing. We investigate this problem both for the setting where we are allowed to permute the votes so as to achieve the single-crossing property and for the setting where the desired ordering of the votes is fixed.

We first focus on the case where the ordering of the votes is provided as part of the input, and show that our problem admits an efficient algorithm when the input profile consists of weak orders, or when no input vote contains an antichain of size 3 and for every pair of candidates there is at least one voter who is able to compare them. We then turn to the problem of checking whether a given profile of partial orders can be extended to a profile of total orders that is single-crossing with respect to *some* ordering of votes. We show that this problem is NP-complete, even if all votes in the input are weak orders with antichains of size at most 2. Given these hardness results, we focus on top orders and obtain polynomial-time algorithms under mild additional assumptions on voters' preferences. We show that, given a profile of top orders $\mathcal{R}$, we can efficiently decide whether it can be extended to a single-crossing profile of total orders if $\mathcal{R}$ contains (i) at least one full vote and (ii) the input profile is *narcissistic*, i.e., each candidate is ranked first by at least one voter.

We also investigate alternative extensions of the single-crossing property to the domain of partial votes. In particular, we define the notion of a *blockwise single-crossing*, and show that such profiles can be detected efficiently.

Relevance of Our Study. We believe that understanding the single-crossing property in the context of partial preference orders is important in its own right. However, our research also has a more direct motivation: knowing that a profile of partial preference orders can be extended to a single-crossing one can simplify the winner determination process, both in single-winner and in multi-winner elections.

Consider a profile of top orders in a single-winner election. If we know that the votes can be extended to a single-crossing profile for a given voter order, then we can find the median vote in this order and pick its top candidate as the winner. This candidate is a possible Condorcet winner and, thus, a natural one to select.

For the case of multi-winner elections, Skowron et al. [41] have shown an efficient winner determination algorithm for the voting rule of Chamberlin and Courant [13], for the case of single-crossing elections (in the general setting, the rule is NP-hard [38, 31]). Their algorithm focuses on the top parts of the votes, but requires the order witnessing that the election is single-crossing. Thus, if we could find an order witnessing that a profile of top orders can be extended to a single-crossing profile, then we could use the algorithm of Skowron et al. [41].

There is a further added benefit of considering the single-crossing property in the context of partial preference orders. Intuitively, when voters cast partial preference orders, they only specify pairwise comparisons that they truly care about. Consequently, the resulting profiles are much more likely to satisfy various structural properties (such as being single-peaked/single-crossing) than profiles where voters are forced to rank candidates that they do not care about (and may therefore rank them in a way that obscures the true preference structure).

Related Work. Both single-peaked and single-crossing preferences can be recognized in polynomial time if the input is a collection of total orders [3, 25, 21, 11]. The problem becomes much more difficult if we ask whether a given preference profile is close to being single-peaked or single-crossing, or, more generally, close to belonging to some restricted domain, for an appropriate notion of distance; indeed, many (though not all) variants of this problem are known to be NP-hard [24, 12, 26]. Both single-peaked and single-crossing preferences arise in societies that are, in some sense, one-dimensional; however, the two notions are distinct, in the sense that there are single-peaked elections that are not single-crossing and vice versa [34]; see also the work of Elkind et al. [22]. Finally, we want to mention dichotomous preferences, a special case of weak orders where voters only distinguish between approved and disapproved candidates. Elkind and Lackner [20] study algorithmic question related to structure in dichotomous preferences and also consider the problem of recognizing single-crossing preferences given a dichotomous input.

2. Preliminaries

For each integer k, we denote the set $\{1, \dots, k\}$ by $[k]$. Let C be a finite set of candidates (alternatives). A *(strict) partial order* is a binary relation $\succ$ over C that has the following properties: for every $a, b, c \in C$ (i) $a \not\succ a$; (ii) $a \succ b$ implies $b \not\succ a$; (iii) $a \succ b$ and $b \succ c$ implies $a \succ c$. We say that a pair of alternatives a, b is *comparable in* $\succ$ if $a \succ b$ or $b \succ a$; otherwise we say that a and b are *incomparable in* $\succ$ and write $a \perp_{\succ} b$. A partial order $\succ$ is said to be *total* if $a \succ b$

or $b \succ a$ for every $a, b \in C$. A total order $\succ$ is an *extension* of a partial order $\succ'$ if for every pair of alternatives a, b such that $a \succ' b$ it holds that $a \succ b$.

When $a \succ b$, we say that $\succ$ ranks a above b. For readability, we will often denote a generic partial order by r and write $a \succ_r b$ or $r : a \succ b$ when r ranks a above b.

A partial order $\succ$ is said to be a *weak order* if for all $a, b, c \in C$ it holds that $a \perp_\succ b$ and $b \perp_\succ c$ implies $a \perp_\succ c$. Equivalently, in a weak order $\succ$ all candidates are partitioned into several equivalence classes $C_1, \dots, C_k$ so that for $a \in C_i$, $b \in C_j$ we have $a \perp_\succ b$ if $i = j$ and $a \succ b$ if $i < j$. Weak orders can be understood as total orders with ties allowed. A *top order* is a weak order with k equivalence classes where $|C_1| = \cdots = |C_{k-1}| = 1$. Intuitively, top orders correspond to a voter ranking some of her most preferred alternatives, and leaving the remaining alternatives unranked. Thus, we refer to the candidates in $\bigcup_{i=1}^{k-1} C_i$ as the *ranked candidates*. A set $C' \subseteq C$ is said to be an *antichain* in $\succ$ if $a \perp_\succ b$ for all $a, b \in C'$.

A list $\mathcal{R} = (r_1, \dots, r_n)$ is called a *profile of partial/weak/top orders* if $r_1, \dots, r_n$ are partial, weak, or top orders, respectively. We refer to elements of $[n]$ as *voters*: the order r_i is the vote of voter i.

Single-Crossing Property. We are now ready to define what it means for a profile to be single-crossing.

Definition 1. *A profile $\mathcal{R} = (r_1, \dots, r_n)$ of total orders over a candidate set C is said to be* single-crossing with respect to a total order $\sqsubset$ on $[n]$ *if for every pair of candidates $a, b \in C$ such that the first voter in $\sqsubset$ prefers a to b it holds that the voters who prefer a to b precede in $\sqsubset$ the voters who prefer b to a. A profile $\mathcal{R} = (r_1, \dots, r_n)$ of total orders is said to be* single-crossing *if there exists a total order $\sqsubset$ on $[n]$ such that $\mathcal{R}$ is single-crossing with respect to $\sqsubset$.*

A natural way to extend this definition to profiles of partial orders is to follow the route taken by Lackner [30] and ask if the partial orders in a given profile can be extended so that the resulting profile of total orders is single-crossing. Following the tradition in the computational social choice literature, which dates back to the seminal paper of Konczak and Lang [29], we refer to such profiles as *possibly single-crossing*.

Definition 2. *A profile $\mathcal{R} = (r_1, \dots, r_n)$ of partial orders over a candidate set C is said to be* possibly single-crossing with respect to a total order $\sqsubset$ on $[n]$ *if there exists a profile $\widehat{\mathcal{R}} = (\widehat{r}_1, \dots, \widehat{r}_n)$ of total orders, where $\widehat{r}_i$ is an extension of r_i for each $i \in [n]$, that is single-crossing with respect to $\sqsubset$. $\mathcal{R}$ is said to be* possibly single-crossing *if there exists a total order* $\sqsubset$ *on* $[n]$ *such that* $\mathcal{R}$ *is single-crossing with respect to* $\sqsubset$*.*

Computational Problems. The goal of this paper is to study the computational complexity of the following two problems (and their special cases):

> PARTIAL ORDER SINGLE-CROSSING CONSISTENCY—FIXED ORDER (PO-SCC-F):
> Given a candidate set C, a profile $\mathcal{R} = (r_1, \dots, r_n)$ of partial orders over C, and a total order $\sqsubset$ on $[n]$, decide whether $\mathcal{R}$ is possibly single-crossing with respect to $\sqsubset$.

> PARTIAL ORDER SINGLE-CROSSING CONSISTENCY (PO-SCC):
> Given a candidate set C and a profile $\mathcal{R} = (r_1, \dots, r_n)$ of partial orders over C, decide whether $\mathcal{R}$ is possibly single-crossing.

We are also interested in special cases of PO-SCC-F and PO-SCC where the input profile contains: (i) weak orders only (WO-SCC-F/WO-SCC), and (ii) top orders only (TO-SCC-F/TO-SCC).

Our primary goal is to distinguish between polynomial-time solvable and NP-complete problems. For some NP-complete problems we present *fixed-parameter tractable (fpt) algorithms*, i.e., algorithms with a runtime of $\mathcal{O}(f(k) \cdot \mathit{poly}(n))$, where k is some parameter of the input, f is a computable function (usually exponential), and n is the input size. Thus, a fixed-parameter tractable algorithm has polynomial runtime if the parameter is fixed to a constant. For an extensive introduction to fpt algorithms and parameterized complexity see the monographs [17, 18].

3. Fixed Order of Votes

To build our intuition concerning the possibly single-crossing property, we start by considering a relaxed variant of this property, which we call the *seemingly single-crossing property*. One advantage of the definition below is that it is easy to check in polynomial time whether a given profile of partial orders is seemingly single-crossing with respect to a given order of voters.

Definition 3. *A profile* $\mathcal{R} = (r_1, \dots, r_n)$ *of partial orders over a candidate set* C *is* seemingly single-crossing with respect to a total order $\sqsubset$ over $[n]$ *if for every pair of candidates* $a, b \in C$ *the voters can be divided into two (possibly empty)* *consecutive intervals with respect to $\sqsubset$ so that (i) in one of these intervals each voter either prefers a to b or indicates that a and b are incomparable, and (ii) in the other interval each voter either prefers b to a or indicates that a and b are incomparable. A profile $\mathcal{R} = (r_1, \dots, r_n)$ of partial orders is* seemingly single-crossing *if it is seemingly single-crossing with respect to some total order $\sqsubset$ over $[n]$.*

By construction, a profile of total orders is single-crossing if and only if it is seemingly single-crossing. Moreover, a profile of partial orders that is possibly single-crossing with respect to a given order of voters is seemingly single-crossing with respect to that order of voters. Indeed, if a profile of partial orders is not seemingly single-crossing with respect to the order of voters $\sqsubset$, there is a pair of distinct alternatives $a, b \in C$ and a triple of voters i, j, k with $i \sqsubset j \sqsubset k$ such that i and k prefer a to b, whereas j prefers b to a; clearly, no refinement of this profile can be single-crossing with respect to $\sqsubset$. The reader may wonder if the converse implication also holds, i.e., whether a profile of partial orders that is seemingly single-crossing with respect to some order of voters $\sqsubset$ can be extended to a profile of total orders that is single-crossing with respect to $\sqsubset$; note that this would immediately imply that PO-SCC-F is polynomial-time solvable. However, the following example shows that this is not the case.

Example 1. Consider the following profile $\mathcal{R} = (r_1, r_2, r_3, r_4)$ of partial orders over the candidate set $C = \{a, b, c\}$

$$r_1\colon a \succ b \succ c, \qquad r_2\colon c \succ b, \qquad r_3\colon b \succ a, \qquad r_4\colon a \succ c.$$

It is easy to see that $\mathcal{R}$ is seemingly single-crossing with respect to the order $1 \sqsubset 2 \sqsubset 3 \sqsubset 4$. However, $\mathcal{R}$ cannot be extended to a profile of total orders $(\widehat{r}_1, \widehat{r}_2, \widehat{r}_3, \widehat{r}_4)$ that is single-crossing with respect to $\sqsubset$. Indeed, $a \succ_{r_1} b$, $b \succ_{r_3} a$ implies that $\widehat{r}_4$ would have to rank b above a, and $b \succ_{r_1} c$, $c \succ_{r_2} b$ means that $\widehat{r}_4$ would have to rank c above b. By transitivity, it follows that $\widehat{r}_4$ ranks c above a, but this is impossible, since $a \succ_{r_4} c$.

This argument does not show that $\mathcal{R}$ is not possibly single-crossing. In fact, $\mathcal{R}$ *is* possibly single-crossing with respect to a different order of voters, namely $1 \sqsubset' 2 \sqsubset' 4 \sqsubset' 3$, as witnessed by the following profile $(\widehat{r}_1, \widehat{r}_2, \widehat{r}_3, \widehat{r}_4)$ of total orders (for convenience, the votes below are listed according to $\sqsubset'$):

$$\widehat{r}_1\colon a \succ b \succ c, \qquad \widehat{r}_2\colon a \succ c \succ b,$$
$$\widehat{r}_4\colon a \succ c \succ b, \qquad \widehat{r}_3\colon c \succ b \succ a.$$

However, we can modify $\mathcal{R}$ so that it remains seemingly single-crossing, but is not possibly single-crossing with respect to *any* order of voters. Specifically, set $C' = \{a, b, c, d, e, f\}$, and consider the following profile $\mathcal{R}' = (r'_1, r'_2, r'_3, r'_4)$ of partial orders over C', which is obtained by prepending a single-crossing profile of total orders over $\{d, e, f\}$ to $\mathcal{R}$:

$$
\begin{aligned}
&r'_1\colon d \succ e \succ f \succ a \succ b \succ c, \\
&r'_2\colon e \succ d \succ f, \qquad f \succ a, \qquad f \succ c \succ b, \\
&r'_3\colon e \succ f \succ d, \qquad d \succ c, \qquad d \succ b \succ a, \\
&r'_4\colon f \succ e \succ d, \qquad d \succ b, \qquad d \succ a \succ c.
\end{aligned}
$$

It is easy to see that $\mathcal{R}'$ is seemingly single-crossing with respect to $1 \sqsubset 2 \sqsubset 3 \sqsubset 4$. Further, the $\{d, e, f\}$-parts of orders in $\mathcal{R}'$ ensure that the only orders for which $\mathcal{R}'$ is seemingly single-crossing are $1 \sqsubset 2 \sqsubset 3 \sqsubset 4$ and $4 \sqsubset 3 \sqsubset 2 \sqsubset 1$. Thus, no extension of $\mathcal{R}$ is single-crossing.

Note that Example 1 uses partial orders that are not weak orders. This is essential: later (Corollary 6), we will see that every seemingly single-crossing profile of weak orders is possibly single-crossing. Thus, we structure the remainder of this section as follows: first, we consider the complexity of PO-SCC-F for arbitrary partial orders, proving an NP-hardness result for the general case and identifying some tractable special cases, and then we focus on weak orders, where we are able to obtain a polynomial-time algorithm for the general case.

3.1. Partial Orders

Example 1 shows that, to solve PO-SCC-F, it is not sufficient to check whether the input profile is seemingly single-crossing, which can be done in polynomial time by verifying this property for every pair of candidates. We were unfortunately not able to determine the complexity of PO-SCC-F. However, we can show that this problem becomes polynomial-time solvable if we additionally assume that no order in the input profile contains an antichain of size 3, and no pair of candidates is incomparable in every vote.

Theorem 2. *One can determine in polynomial time whether a profile of partial orders $\mathcal{R}$ is possibly single-crossing with respect to a given order $\sqsubset$ on $[n]$ if both of the following two conditions hold:*

(1) $\mathcal{R}$ does not contain a vote with an antichain of size 3, and
(2) no pair of candidates is incomparable in every vote.

Proof. We reduce our problem to the Boolean satisfiability problem with clause size two. Consider a candidate set $C = \{c_1, \ldots, c_m\}$, and let $\sqsubset$ be the given order on $[n]$. For each pair of distinct candidates c_i, c_j and each vote r we define a Boolean variable named $[r : c_i \succ c_j]$, representing the truth of the statement "$r : c_i \succ c_j$".

The following conditions have to hold in order for $\mathcal{R}$ to be possibly single-crossing. For every vote $r \in \mathcal{R}$ and every triple of distinct candidates c_i, c_j, c_k it should hold that:

$$[r : c_i \succ c_j] \leftrightarrow \neg[r : c_j \succ c_i] \quad \text{(antisymmetry)},$$

$$[r : c_i \succ c_j] \wedge [r : c_j \succ c_k] \rightarrow [r : c_i \succ c_k] \quad \text{(transitivity)},$$

and for every triple of votes r, r', r'' with $r \sqsubset r' \sqsubset r''$ it should hold that

$$\neg([r : c_i \succ c_j] \wedge [r' : c_j \succ c_i] \wedge [r'' : c_i \succ c_j]) \quad \text{(single-crossing)}.$$

Our aim is to show that when conditions (1) and (2) in the theorem statement are satisfied, the resulting Boolean formula can be represented as a formula with clause size two. Since Boolean satisfiability for formulas with clause size two is decidable in linear time [2], we require only polynomial time.

First, the clauses enforcing antisymmetry already have size two. For any transitivity clause, at least one of the variables is already known to us, as otherwise we would have an antichain of size three. We can replace this variable with a fixed truth value, obtaining a clause of size two.

Let us now consider the single-crossing property. Recall that for every pair of candidates c_i, c_j there is at least one vote r^+ in which these two candidates are comparable. We distinguish two cases: $r^+ : c_i \succ c_j$ and $r^+ : c_j \succ c_i$.

First, we assume that $[r^+ : c_i \succ c_j]$ is true. We have to consider the relative positions of r^+, r, r', r'' in $\sqsubset$. If r^+ is to the left of r', i.e., either $r^+ \sqsubset r \sqsubset r' \sqsubset r''$ or $r \sqsubset r^+ \sqsubset r' \sqsubset r''$ holds, we can replace the single-crossing condition for r, r', r'' by $\neg([r' : c_j \succ c_i] \wedge [r'' : c_i \succ c_j])$. Indeed, if the modified condition is violated, the profile is not possibly single-crossing with respect to $\sqsubset$, as witnessed by $r^+ : c_i \succ c_j$, $r' : c_j \succ c_i$ and $r'' : c_i \succ c_j$ (and if the modified condition is satisfied, so is the original condition). If r^+ is to the right of r', we can use the same argument to drop r'' from the formula.

Let us now assume that $[r^+ : c_j \succ c_i]$ is true. Again, we have to consider the position of r^+ in $\sqsubset$. If r^+ is to the left of r, i.e., $r^+ \sqsubset r \sqsubset r' \sqsubset r''$, we can replace the single-crossing condition for r, r', r'' by $\neg([r : c_i \succ c_j] \wedge [r' : c_j \succ c_i])$: if the modified condition is violated, the profile is not possibly single-crossing with respect to $\sqsubset$, as witnessed by the triple of voters r^+, r, r' and the pair of candidates c_i, c_j. A similar argument applies if r^+ is to the right of r''. Finally, if r^+ is between r and r'', we can replace the single-crossing condition for r, r', r'' by $\neg([r : c_i \succ c_j] \wedge [r'' : c_i \succ c_j])$. □

If a profile does not satisfy conditions (1) and (2) in the statement of Theorem 2, but there are only few antichains of size 3 and 'universally' incomparable candidate pairs, we can still solve PO-SCC-F efficiently.

Theorem 3. *Let a denote the total number of antichains of size 3 in $\mathcal{R}$ and let b denote the number of candidate pairs that are incomparable in all votes. It is possible to determine whether a profile of partial orders $\mathcal{R}$ is possibly single-crossing with respect to a given order in time $\mathcal{O}\left(2^{a+b} \cdot poly(m,n)\right)$.*

Proof. We use the satisfiability encoding of Theorem 2. We guess one variable of each antichain and, for each pair of candidates that are incomparable in every vote, we guess the preferences of the first voter over these candidates. For each collection of guesses, we obtain an instance that satisfies the two conditions of Theorem 2; it remains to observe that there are 2^{a+b} guesses to be considered. □

3.2. Weak Orders

For weak orders, a stronger result is true: WO-SCC-F is polynomial-time solvable with no additional constraints on the input profile. Moreover, for weak orders the phenomenon illustrated in Example 1 does not arise: every seemingly single-crossing profile of weak orders is possibly single-crossing. To prove this, we will now present an algorithm that, given a profile $\mathcal{R}$ of weak orders that is seemingly single-crossing with respect to an ordering $\sqsubset$, explicitly constructs an extension of $\mathcal{R}$ that is single-crossing with respect to $\sqsubset$. We first describe a subroutine $\mathcal{E}$ used by our algorithm.

Algorithm $\mathcal{E}$: The algorithm takes as input a profile $\mathcal{R} = (r_1, \ldots, r_n)$ of weak orders, where r_1 is a total order, and an order $\sqsubset$ over $[n]$ such that $1 \sqsubset i$ for each $i \in \{2, \ldots, n\}$. It computes a profile of total orders as follows:

1. It orders the votes in $\mathcal{R}$ according to $\sqsubset$ to obtain a profile $\mathcal{S} = (s_1, \ldots, s_n)$; note that $r_1 = s_1$.
2. It sets $\widehat{s}_1 = s_1$ and for each $i \in \{2, \ldots, n\}$ (in the ascending order), it extends s_i to $\widehat{s}_i$ by ranking all the unranked candidates in maximal antichains the same way as in $\widehat{s}_{i-1}$ (note that by the time it processes s_i, $\widehat{s}_{i-1}$ is a total order).

3. It returns $(\widehat{s}_1, \dots, \widehat{s}_n)$.

Theorem 4. *There is a polynomial-time algorithm that given a profile $\mathcal{R}$ of weak orders that is seemingly single-crossing with respect to an order $\sqsubset$ on $[n]$, outputs an extension of $\mathcal{R}$ that is single-crossing.*

Proof. Let C be a set of candidates and let $\mathcal{R} = (r_1, \dots, r_n)$ be a profile of weak orders over C that is seemingly single-crossing with respect to an order $\sqsubset$ on $[n]$. Without loss of generality, we assume that $\sqsubset$ is given by $1 \sqsubset 2 \sqsubset \cdots \sqsubset n$. To find a single-crossing extension $\widehat{\mathcal{R}} = (\widehat{r}_1, \dots, \widehat{r}_n)$ of $\mathcal{R}$, we first compute an extension $\widehat{r}_1$ of r_1:

1. Set $\widehat{r}_1 = r_1$.
2. Repeat the following steps until $\widehat{r}_1$ is a total order:
   (a) Pick a maximal antichain in $\widehat{r}_1$. Let $C' \subseteq C$ be the corresponding set of candidates.
   (b) For each $i = 2, \dots, n$, if C' is not an antichain in r_i (i.e., r_i ranks some candidates in C'), then order C' in r_1 the same way as in r_i.
   (c) If C' is an antichain in all votes, order C' in r_1 arbitrarily.

Now we have a profile $\mathcal{R}' = (\widehat{r}_1, r_2, \dots, r_n)$ of top orders, where $\widehat{r}_1$ is a total order. We run Algorithm $\mathcal{E}$ on $\mathcal{R}'$ to obtain a profile of total orders. Note that in Algorithm $\mathcal{E}$ we set $\mathcal{S} = \mathcal{R}'$, and therefore this profile, which we will denote by $\widehat{\mathcal{R}} = (\widehat{r}_1, \widehat{r}_2, \dots, \widehat{r}_n)$, is an extension of $\mathcal{R}$. We claim that $\widehat{\mathcal{R}}$ is single-crossing with respect to $\sqsubset$.

Suppose that $\widehat{\mathcal{R}}$ is not single-crossing and let ℓ be the largest index such that $(\widehat{r}_1, \dots, \widehat{r}_{\ell-1})$ is single-crossing. Thus, $(\widehat{r}_1, \dots, \widehat{r}_\ell)$ is not single-crossing and there exists a pair a, b of candidates such that

$$\widehat{r}_1 \colon a \succ b, \qquad \widehat{r}_{\ell-1} \colon b \succ a, \qquad \widehat{r}_\ell \colon a \succ b.$$

(Although the single-crossing property is violated if the conditions above hold for some triple $\widehat{r}_x, \widehat{r}_y, \widehat{r}_z$ with $x < y < z$, due to the minimality of ℓ we can assume that $x = 1$, $y = \ell - 1$, $z = \ell$.) Candidates a and b are ranked differently in $\widehat{r}_{\ell-1}$ and $\widehat{r}_\ell$, so Algorithm $\mathcal{E}$ could not have derived the ranking $a \succ b$ in $\widehat{r}_\ell$ from $\widehat{r}_{\ell-1}$. Hence, in r_ℓ we also have $a \succ b$. Since $\widehat{r}_1$ and $\widehat{r}_{\ell-1}$ rank a and b differently and given how vote $\widehat{r}_1$ is computed, there must be a k, $1 \leq k < \ell - 1$ such that $r_k \colon a \succ b$ and neither a nor b are ranked in any r_i, $i \in [k-1]$. Moreover, by the same argument, there must be a k', $k < k' < \ell$ such that $r_{k'} \colon b \succ a$ and neither a nor b are ranked in any r_i, $k' < i < \ell$. Consequently, the triple $(r_k, r_{k'}, r_\ell)$ witnesses that $\mathcal{R}$ is not seemingly single-crossing with respect to $1 \sqsubset 2 \sqsubset \cdots \sqsubset n$, a contradiction with our assumption. Thus, the algorithm outputs a single-crossing extension of $\mathcal{R}$. □

This theorem has two important consequences.

Corollary 5. WO-SCC-F *is solvable in polynomial time.*

Proof. If the given profile $\mathcal{R}$ of weak orders is seemingly single-crossing with respect to the order $\sqsubset$, we can apply the algorithm of Theorem 4. If $\mathcal{R}$ is not seemingly single-crossing with respect to $\sqsubset$, then it cannot be possibly single-crossing with respect to $\sqsubset$. □

Corollary 6. *A profile of weak order is possibly single-crossing if and only if it is seemingly single-crossing.*

4. Arbitrary Order of Voters

We will now consider the scenario where the ordering of the votes is not given in the input, and we have to decide whether the given profile is possibly single-crossing with respect to *some* ordering of the votes. Note that in this setting we can assume that all votes in the input profile $\mathcal{R}$ are pairwise distinct, as we can simply remove all duplicates without changing the answer. Therefore, we can view $\mathcal{R}$ as a *set* of votes, and identify a voter i with her vote r_i. In particular, it will sometimes be convenient to write $r_i \sqsubset r_j$ in place of $i \sqsubset j$.

4.1. Partial Orders and Weak Orders

In the last section, we have seen that PO-SCC-F is NP-complete. If the ordering of the votes is not part of the input, i.e., we consider PO-SCC, hardness even extends to weak orders. To show this, we will provide a reduction from the BETWEENNESS problem, defined below, which is known to be NP-complete [37].

> BETWEENNESS:
> Given a set $S = \{s_1, \dots, s_m\}$ and a set T of triples over S, decide whether there exists a total order $<$ over S such that for each triple (s_i, s_j, s_k) in T it holds that either $s_i < s_j < s_k$ or $s_k < s_j < s_i$.

To reduce BETWEENNESS to PO-SCC, we use instances of the following gadget. Let $\mathcal{R} = (r_1, r_2, r_3)$ be a profile over candidate set $\{a, b, c, d\}$, where:

$$r_1\colon a \succ b \succ c \succ d,$$
$$r_2\colon b \succ a \succ c \succ d,$$
$$r_3\colon b \succ a \succ d \succ c.$$

The reader can verify that this profile is single-crossing only with respect to the order $1 \sqsubset 2 \sqsubset 3$ and its reverse.

Theorem 7. WO-SCC *problem is* NP*-complete, even for votes with antichains of size at most 2.*

Proof. Clearly, this problem is in NP. To show that is is NP-hard, we provide a reduction from BETWEENNESS.

Let $I = (S, T)$ be an instance of BETWEENNESS, where $S = (s_1, \dots, s_m)$ and $T = (t_1, \dots, t_n)$ is a set of triples over S. The idea of our proof is to form a profile where the voters correspond to the elements of the set S and the constraints from the set T are implemented within the partial orders using the gadget described just before the theorem statement. We let $E = A \cup B \cup C$, where $A = \{a_1, \dots, a_n\}$, $B = \{b_1, \dots, b_n\}$, $C = \{c_1, \dots, c_n\}$, and $D = \{d_1, \dots, d_n\}$, and form a profile $\mathcal{R} = (r_1, \dots, r_m)$ of partial orders over E as follows:

1. For each $\ell \in [m]$, each $i, j \in [n]$, $i < j$, each $x \in \{a_i, b_i, c_i, d_i\}$ and each $y \in \{a_j, b_j, c_j, d_j\}$, we set $r_\ell : x \succ y$.
2. For each triple $t_\ell = (s_i, s_j, s_k) \in T$, we set:

$$r_i\colon a_\ell \succ b_\ell \succ c_\ell \succ d_\ell,\ \ r_j\colon b_\ell \succ a_\ell \succ c_\ell \succ d_\ell,\ \ r_k\colon b_\ell \succ a_\ell \succ d_\ell \succ c_\ell.$$

3. Finally, for each $\ell \in [m]$ and $i \in [n]$, $r_i : a_\ell \succ c_\ell$, $r_i : b_\ell \succ c_\ell$, $r_i : a_\ell \succ d_\ell$, and $r_i : b_\ell \succ d_\ell$.

Observe that two candidates $x, y \in E$ are only incomparable if they share the same index and either $x \in A$ and $y \in B$ or if $x \in C$ and $y \in D$. Thus, we have defined a weak order with antichains of size at most 2.

We claim that $\mathcal{R}$ is possibly single-crossing if and only if I is a "yes"-instance of BETWEENNESS. First, assume that $\mathcal{R}$ is possibly single-crossing with respect to some order $\sqsubset$. By construction, for each triple $t_\ell = (s_i, s_j, s_k) \in T$, we have either $r_i \sqsubset r_j \sqsubset r_k$ or $r_k \sqsubset r_j \sqsubset r_i$. This means that an order $<$ over S such that $s_x < s_y$ if and only if $r_x \sqsubset r_y$ witnesses that I is a "yes"-instance of the BETWEENNESS problem.

On the other hand, assume that I is a "yes"-instance of the BETWEENNESS problem and that some order $<$ over S witnesses this. We define an order $\sqsubset$ over $\{r_1, \ldots, r_m\}$ so that $r_x \sqsubset r_y$ if and only if $s_x < s_y$. To show that $\mathcal{R}$ is possibly single-crossing with respect to $\sqsubset$, we will now extend $\mathcal{R}$ to a profile of total orders as follows. Consider a triple $t_\ell = (s_i, s_j, s_k) \in T$. Without loss of generality, assume that $s_i < s_j < s_k$ (the case $s_k < s_j < s_i$ can be handled in a similar way). We define the voters' preferences regarding a_ℓ, b_ℓ, c_ℓ as follows:

1. For each r_x such that $r_x \sqsubset r_i$, set $r_x : a_\ell \succ b_\ell \succ c_\ell \succ d_\ell$.
2. For each r_y ($y \neq j$) such that $r_i \sqsubset r_y \sqsubset r_k$, set $r_y : b_\ell \succ a_\ell \succ c_\ell \succ d_\ell$.
3. For each r_z such that $r_k \sqsubset r_z$, set $r_z : b_\ell \succ a_\ell \succ d_\ell \succ c_\ell$.

After this operation, the profile consists of total orders, and it is clear that it is single-crossing with respect to $\sqsubset$. □

We have just seen that WO-SCC is NP-complete in general. In the following we will present a fixed-parameter algorithm that is particularly efficient for profiles of weak orders that are close to being total orders. More formally, we consider profiles the maximum number of incomparable pairs in each of the votes. The fpt algorithm is based on the following proposition.

Proposition 8. *There is a polynomial-time algorithm that given an instance $I = (C, \mathcal{R})$ of* WO-SCC*, where $\mathcal{R} = (r_1, \ldots, r_n)$ is a profile of weak orders, and an index ℓ such that r_ℓ is a total order, decides if there is an order $\sqsubset$ such that: (i) for each k, $k \neq \ell$, $r_\ell \sqsubset r_k$, and (ii) $\mathcal{R}$ is possibly single-crossing with respect to $\sqsubset$.*

Proof. Without loss of generality, we can assume that $\ell = 1$. Our algorithm consists of two parts. First, in Algorithm $\mathcal{L}$, we compute an order $\sqsubset$ witnessing that $\mathcal{R}$ is seemingly single-crossing (if indeed it is), and then we invoke Algorithm $\mathcal{E}$ to compute an appropriate extension of $\mathcal{R}$. If Algorithm $\mathcal{L}$ fails at any point, we reject the input (if we reach Algorithm $\mathcal{E}$, failure is impossible).

By the theorem's assumptions, the first element in $\sqsubset$, r_1, is a total order. We define a relation $\sqsubset^*$ over $\{r_2, \ldots, r_n\}$ as follows: For each i, j, $2 \leq i, j \leq n$, if there is a pair of candidates $a, b \in C$ such that r_1 and r_i order a, b identically but r_j orders them differently, we set $r_i \sqsubset^* r_j$. Algorithm $\mathcal{L}$ is given below:

Algorithm $\mathcal{L}$: We compute the relation $\sqsubset^*$ over $\{r_2, \ldots, r_n\}$ and extend it to relation $\sqsubset^{**}$ over $\mathcal{R}$ as follows: for each pair $i, j \in [n]$ we set $r_i \sqsubset^{**} r_j$ if either $i = 1$ or $r_i \sqsubset^* r_j$. Using the standard algorithm for topological sorting, we check if $\sqsubset^{**}$ can be extended to a linear order. If so, we compute and return this order (this will be our order $\sqsubset$). If not, we reject.

It is immediate that if this algorithm rejects then $\mathcal{R}$ is not possibly single-crossing with respect to any order $\sqsubset$ that places r_1 first. We claim that if it does not reject, then the profile $\mathcal{R}$ is seemingly single-crossing with respect to the order $\sqsubset$ computed by $\mathcal{L}$. If it were not, then there would be two candidates a and b and two integers k and ℓ, $1 < k, \ell \leq n, k \neq n$, such that $r_1 \sqsubset r_k \sqsubset r_\ell$ and $a \succ_{r_1} b, b \succ_{r_k} a$, and $a \succ_{r_\ell} b$. However, by definition of $\sqsubset^*$, we would have $r_\ell \sqsubset^* r_k$, contradicting the fact that Algorithm $\mathcal{L}$ did not reject. Thus, $\mathcal{R}$ is seemingly single-crossing with respect to $\sqsubset$. Now, by Theorem 4, we can invoke Algorithm $\mathcal{E}$ with $\mathcal{R}$ and $\sqsubset$ as input to get a single-crossing extension of $\mathcal{R}$. $\square$

Let k-WO-SCC be the special case of WO-SCC where each vote contains at most k incomparable pairs of candidates.

Theorem 9. *The* k-WO-SCC *problem can be decided in time* $\mathcal{O}(2^k \cdot poly(m, n))$.

Proof. Let $\mathcal{R} = (r_1, \ldots, r_n)$ be our input profile. For each r_i we execute the following steps (if we accept for some r_i, then we accept the whole input, and if we fail for each r_i, then we reject the whole input):

1. Compute the set R_i of all possible extensions of r_i to a total order (there are at most 2^k of them).
2. For each r in R_i, we execute the following:
   (a) We apply Algorithm $\mathcal{L}$ from the proof of Proposition 8 to the profile $(r, r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_n)$ with r taking the role of r_1 in this algorithm. Note that Algorithm $\mathcal{L}$ simply requires that r is a total order and does not require the rest of the profile to consist of top orders. Let $\sqsubset$ be the output of $\mathcal{L}$. If this operation succeeds, we proceed to the next step.
   (b) We use a variant of Algorithm $\mathcal{E}$ to compute the extension $\widehat{\mathcal{R}}$ of $\mathcal{R}$ (namely, we extend all the votes one by one in the order $\sqsubset$, setting the order of each incomparable pair in the current vote to be the same as in the preceding, just extended vote). The algorithm terminates with $\widehat{\mathcal{R}}$ as the single-crossing extension of $\mathcal{R}$.

It is straightforward to see that the algorithm is correct: In essence, it performs a full search through all reasonable choices of the order $\sqsubset$ and uses only "safe" extensions of votes in the modification of Algorithm $\mathcal{E}$. $\square$

4.2. Top Orders

The case of top orders is by far the most important and practical one. It turns out that it is also quite challenging: we have not been able to determine the exact complexity of TO-SCC. Nonetheless, we will now describe a polynomial-time algorithms for this problem that works under additional mild constraints on voters' preferences.

For this result, we need to assume that our profile of top orders is *narcissistic*, i.e., every candidate is ranked first by at least one voter; this assumption dates back to the work of Bartholdi and Trick [3], and has been used in several recent computational social choice papers [15, 41]; we expect it to be satisfied when candidates are allowed to vote in the election. For such profiles, we can relax the condition of Proposition 8: we still require that at least one voter submits a total order, but make no assumptions about this voter's position in the profile. We remark that one can assume that the profile contains a total order if, e.g., the person who wants to understand if the given election is possibly single-crossing is herself a voter in this election; this assumption is also required for some of the algorithmic results of Lackner [30], and is known to reduce the communication complexity for vote elicitation [14].

Theorem 10. *There is a polynomial-time algorithm that given an instance* $I = (C, \mathcal{R})$ *of the* Top Partial Order SCC *problem, where* $\mathcal{R} = (r_1, \dots, r_n)$ *is narcissistic and contains at least one total order, decides if* $\mathcal{R}$ *is possibly single-crossing.*

Proof. In what follows, it will be convenient to assume that no top order in $\mathcal{R}$ is a prefix of another top order in $\mathcal{R}$, i.e., it cannot be the case that $r_i = c_{i_1} \succ \dots \succ c_{i_k}$, $r_j = c_{i_1} \succ \dots \succ c_{i_\ell}$ for some $i, j \in [n]$, $i \neq j$, and $k, \ell \in [m]$. Indeed, if such a pair of votes is present, we can remove the "shorter" vote; the modified profile is a "yes"-instance of our problem if and only if the original profile was.

We will use the fact that every narcissistic single-crossing profile is single-peaked [22]. We remind the reader that a vote r over a candidate set C is said to be *single-peaked with respect to an sp-axis* $\lhd$, where $\lhd$ is a total order over C, if for every $j \leq |C|$ the set of candidates ranked in top j positions in r forms a contiguous segment of $\lhd$; a profile $\mathcal{R}$ is said to be *single-peaked with respect to* $\lhd$ if every vote in $\mathcal{R}$ is single-peaked with respect to $\lhd$ (among the many equivalent definitions of single-peaked preferences, this one is best suited for our purposes).

Let $C = \{c_1, \dots, c_m\}$, and let $\mathcal{R}(c)$ denote the set of all votes in $\mathcal{R}$ that rank c first. We say that a total order $\sqsubset$ of voters and a total order $\lhd$ of candidates are *consistent* if for every pair of candidates $a, b \in C$ it holds that $a \lhd b$ implies that in $\sqsubset$ all votes in $\mathcal{R}(a)$ precede all votes in $\mathcal{R}(b)$. It is immediate from Proposition 5 in [22] that if a narcissistic profile $\mathcal{R}$ is single-crossing with respect to an ordering of votes given by $\sqsubset$, then the (unique) order of candidates $\lhd$ that is consistent with it is an sp-axis witnessing that $\mathcal{R}$ is single-peaked; we remark that $\lhd$ is well-defined since for every candidate $a \in C$ the votes in $\mathcal{R}(a)$ form a contiguous segment of $\sqsubset$. Another useful fact is that, if all votes in a single-crossing profile are distinct, the ordering of votes that witnesses that this profile is single-crossing is unique up to a reversal (see Proposition 1 in [19]). Consequently the respective order $\lhd$ is unique up to a reversal as well.

The algorithm proceeds in two stages. First, it builds up the two "ends" of the voters' order, moving towards the center. It uses several rules to extend these ends, which will be detailed below. When none of these rules are applicable, there is a clear separation between the candidates in the already-ordered votes and the remaining candidates. We then build a smaller instance of our problem that involves only a subset of candidates and a subset of votes, call our algorithm recursively on this smaller instance, and then check if the result can be "pasted" into the original profile. We will now present the details of the algorithm.

We will say that an ordering $\sqsubset$ of the votes in $\mathcal{R}$ is *good* if $\mathcal{R}$ can be extended to a complete profile that is single-crossing with respect to $\sqsubset$ and all votes in $\mathcal{R}(c_1)$ precede in $\sqsubset$ all votes in $\mathcal{R}(c_m)$. Also, we will say that a total order $\lhd$ on C is *good* if it is consistent with some good ordering of the votes; note that this implies that $c_1 \lhd c_m$ and there is a complete extension of $\mathcal{R}$ that is single-peaked with respect to $\lhd$. We note that $(C, \mathcal{R})$ is a "yes"-instance of our problem if and only if there exists a good ordering of the votes in $\mathcal{R}$; indeed, we can always ensure that c_1 precedes c_m by reversing the ordering, if necessary.

Consider our input profile $\mathcal{R} = (r_1, \ldots, r_n)$. We will build two total orders $\lhd_L$ (for "left") and $\lhd_R$ (for "right") on two disjoint sets of candidates $C_L \subseteq C$ and $C_R \subseteq C$. When doing so, we will ensure that every good order $\lhd$ on C coincides with $\lhd_L$ on C_L and with $\lhd_R$ on C_R, and, moreover, $c \lhd c'$ whenever $c \in C_L$, $c' \in C \setminus C_L$ or $c \in C \setminus C_R$, $c' \in C_R$. In a similar vein, we build two disjoint sets of votes $\mathcal{R}_L \subseteq \mathcal{R}$ and $\mathcal{R}_R \subseteq \mathcal{R}$ and two total orders $\sqsubset_L$ and $\sqsubset_R$ on these sets so that every good ordering $\sqsubset$ of the votes in $\mathcal{R}$ coincides with $\sqsubset_L$ on $\mathcal{R}_L$ and with $\sqsubset_R$ on $\mathcal{R}_R$ and, moreover, $r \sqsubset r'$ whenever $r \in \mathcal{R}_L$, $r' \in \mathcal{R} \setminus \mathcal{R}_L$ or $r \in \mathcal{R} \setminus \mathcal{R}_R$, $r' \in \mathcal{R}_R$. We denote the "rightmost" candidate in C_L by c^L: formally, $c^L \in C_L$ and $c \lhd c^L$ for all $c \in C_L \setminus \{c^L\}$. Similarly, c^R is the "leftmost" candidate in C_R: $c^R \in C_R$ and $c^R \lhd c$ for all $c \in C_R \setminus \{c^R\}$. We also set $C_M = C \setminus (C_L \cup C_R)$, $\mathcal{R}_M = \mathcal{R} \setminus (\mathcal{R}_L \cup \mathcal{R}_R)$. Initially we have $C_M = C$, $C_L = C_R = \emptyset$ and $\mathcal{R}_M = \mathcal{R}$,

$\mathcal{R}_L = \mathcal{R}_R = \emptyset$. For readability, we will sometimes view the partial orders $\lhd_L$, $\lhd_R$, $\sqsubset_L$, and $\sqsubset_R$ as sequences of elements of C_L, C_R, $\mathcal{R}_L$, and $\mathcal{R}_R$, respectively.

Let r^* be some total order that appears in $\mathcal{R}$, and assume that r^* is given by $c_m \succ \cdots \succ c_1$. Since r^* ranks c_1 last, for every good axis $\lhd$ it holds that c_1 is the leftmost candidate with respect to $\lhd$. Hence we set $C_L := \{c_1\}$, $C_M := C_M \setminus \{c_1\}$, $\lhd_L = (c_1)$, $c^L = c_1$.

We will now present several rules that enable us to extend the orders $\lhd_L$, $\lhd_R$, $\sqsubset_L$, and $\sqsubset_R$, and grow the sets C_L, C_R, $\mathcal{R}_L$, and $\mathcal{R}_R$.

(L1) Consider a voter r that ranks c_1 first. Suppose that r contains candidates other than c_1, i.e., $r = c_1 \succ c_{i_1} \succ \cdots \succ c_{i_k}$. If a total order $\lhd$ satisfies $c_1 \lhd c_m$, but does not have $(c_1, c_{i_1}, \ldots, c_{i_k})$ as a prefix, then any extension of r is not single-peaked with respect to $\lhd$. Thus, if $\lhd$ is good, it has $(c_1, c_{i_1}, \ldots, c_{i_k})$ as a prefix, and hence we can extend C_L and $\lhd_L$ by setting $C_L = C_L \cup \{c_{i_1}, \ldots, c_{i_k}\}$, $\lhd_L = (c_1, c_{i_1}, \ldots, c_{i_k})$. This also means that if we have a "yes"-instance of our problem, the vote that ranks c_1 first must be unique (recall that we assume that all votes are distinct and no vote is a proper prefix of another vote). Thus, if a vote that ranks c_1 first is not unique, we stop and output "no". Otherwise we use the unique vote r that ranks c_1 first to extend C_L and $\lhd_L$, as described above, and set $C_M = C_M \setminus \{c_{i_1}, \ldots, c_{i_k}\}$, $c^L = c_{i_k}$.

(L2) Suppose that some candidate $c \in C_L$ appears in $\lhd_L$, for each candidate $c' \in C_L$ with $c' \lhd_L c$ the votes in $\mathcal{R}(c')$ have been placed in $\sqsubset_L$, but none of the votes in $\mathcal{R}(c)$ has been placed in $\sqsubset_L$ yet. Consider some good order $\sqsubset$. As $\sqsubset$ is consistent with some good $\lhd$, and every good $\lhd$ has $\lhd_L$ as a prefix, $\mathcal{R}(c)$ needs to be appended to the right end of $\sqsubset_L$. Now, consider two votes $r', r'' \in \mathcal{R}(c)$, and let i be the first position where they differ; let c' be the candidate ranked i-th in r', and let c'' be the candidate ranked i-th in r''. Extensions of r' and r'' are single-peaked with respect to some good order $\lhd$, and $\lhd_L$ is a prefix of any such $\lhd$. Thus, if we have a "yes"-instance of our problem, for any good order $\lhd$ it holds that $c' \lhd c \lhd c''$ or $c'' \lhd c \lhd c'$; assume without loss of generality that $c' \lhd c \lhd c''$. Then the votes from $\mathcal{R}(c'')$ appear after r' and r'' in every good $\sqsubset$. This means that r' appears before r'' in $\sqsubset$, as otherwise c' and c'' would cross more than once. In this way, we can decide on the relative order of all votes in $\mathcal{R}(c)$; if the resulting relation on these votes turns out to be cyclic, we output "no" and stop. Otherwise, we set $\mathcal{R}_L := \mathcal{R}_L \cup \mathcal{R}(c)$ and append the votes in $\mathcal{R}(c)$ to $\sqsubset_L$ in the prescribed order. Finally, we check that $\mathcal{R}_L$ is seemingly single-crossing with respect to $\sqsubset_L$, and output "no" and stop if this is not the case.

(L3) Now, suppose that there is a vote $r \in \mathcal{R}_L$ and a candidate that appears in this vote, but has not been placed in $\lhd_L$. Let c be the highest-ranked candidate in r with this property. For every good order $\lhd$ it holds that the candidates appearing before c in r form a contiguous segment of $\lhd$. Therefore, we append c to the right end of $\lhd_L$. We also set $C_L := C_L \cup \{c\}$, $C_M := C_M \setminus \{c\}$, $c^L := c$.

We apply rules (L2) and (L3) until neither of them is applicable: all candidates ranked by voters in $\mathcal{R}_L$ appear in $\lhd_L$ and all votes that rank a candidate in $\lhd_L$ first have been added to $\mathcal{R}_L$. At this point, we have $\lhd_L = c_1 \lhd_L \cdots \lhd_L c^L$. We now turn our attention to the total order r^*. If $r^* \in \mathcal{R}_L$, then $\lhd_L$ is a total order over C and hence all voters from $\mathcal{R}$ have been placed $\mathcal{R}_L$; since $\mathcal{R}_L$ is seemingly single-crossing, we can invoke Theorem 4 to produce a total extension of $\mathcal{R}_L$. Thus, assume that $r^* \notin \mathcal{R}_L$, i.e., on every good axis c_m appears after all votes in C_L.

(M1) We first check that r^* agrees with $\lhd_L$, i.e., if $a, b \in C_L$ and $a \lhd_L b$, then r^* ranks b above a. If this is not the case, we output "no" and stop. Now, suppose that some candidates from $C \setminus C_L$ appear in r^* below c^L; let $c_{i_1}, \dots, c_{i_k}$ be the list of all such candidates, listed in order of their appearance in r^* (i.e., $c_{i_1} \succ_{r^*} \cdots \succ_{r^*} c_{i_k}$). Consider an arbitrary good $\lhd$. As r^* has to be single-peaked with respect to $\lhd$, it follows that $c_{i_1}, \dots, c_{i_k}$ appear on the right end of $\lhd$, in this order. Hence, we set $C_R := \{c_{i_1}, \dots, c_{i_k}\}$, $\lhd_R := (c_{i_1}, \dots, c_{i_k})$, $C_M := C_M \setminus \{c_{i_1}, \dots, c_{i_k}\}$, $c^R := c_{i_1}$.

We will now build up C_R, $\lhd_R$, $\mathcal{R}_R$, and $\sqsubset_R$ in the same way as C_L, $\lhd_L$, $\mathcal{R}_L$, and $\sqsubset_L$, i.e., we define rules (R2) and (R3) by making appropriate modifications to (L2) and (L3), and apply them for as long as they extend $\lhd_R$ or $\sqsubset_R$. Once we get stuck (and $\mathcal{R}_M \neq \emptyset$), we consult with r^* again, using the following generalization of (M1).

(M2) We first check that r^* agrees with $\lhd_L$ and $\lhd_R$, i.e., if $a, b \in C_L$ and $a \lhd_L b$, then r^* ranks b above a, and if $a, b \in C_R$ and $a \lhd_R b$, then r^* ranks a above b. If this is not the case, we output "no" and stop. Now, suppose that some candidates from C_M appear in r^* below some candidates in $C_L \cup C_R$; let $C' = \{c \in C_M \mid c' \succ_{r^*} c \text{ for some } c' \in C_L \cup C_R\}$. If there exist candidates $c \in C_M$, $c' \in C_L$, $c'' \in C_R$ such that r^* ranks both c' and c''

above c, we output "no" and stop, as this means that r^* is not single-peaked with respect to any axis $\lhd$ that has $\lhd_L$ as its prefix and $\lhd_R$ as its suffix. Hence, we can assume that all candidates in the set $\{c \in C_L \cup C_R \mid c \succ_{r^*} c' \text{ for some } c' \in C'\}$ belong to exactly one of the sets C_L, C_R. Suppose that this set is C_R, i.e., some $c_j \in C_R$ is the highest-ranked candidate from $C_L \cup C_R$ in r^*, and let $c_{i_1}, \dots, c_{i_k}$ be the list of all candidates from C_M that appear below c_j in r^*, listed in order of their appearance in r^* (i.e., $c_{i_1} \succ_{r^*} \cdots \succ_{r^*} c_{i_k}$). Consider an arbitrary good order $\lhd$. As r^* has to be single-peaked with respect to $\lhd$, it follows that $c_{i_k}, \dots, c_{i_1}$ appear just after c^L in $\lhd$, in this order. Therefore, we set $C_L := C_L \cup \{c_{i_1}, \dots, c_{i_k}\}$ and extend $\lhd_L$ by appending $(c_{i_k}, \dots, c_{i_1})$ to it; also, we set $C^L := c_{i_1}$. By the same argument, if the highest-ranked candidate from $C_L \cup C_R$ in r^* belongs to C_L, we let $C_R := C_R \cup \{c_{i_1}, \dots, c_{i_k}\}$ and extend $\lhd_R$ by prepending $(c_{i_1}, \dots, c_{i_k})$ to it; also, we set $C^R := c_{i_1}$. Finally, we remove $\{c_{i_1}, \dots, c_{i_k}\}$ from C_M.

After (M2) has been applied, we can try to apply (L2) and (L3) again. If this extends $\lhd_L$, (M2) may be applicable again, producing an extension of $\lhd_R$, so (R2) and (R3) may become applicable. We proceed in this manner until none of our rules applies. This means, in particular, that r^* ranks all candidates in C_M above all candidates in $C_L \cup C_R$. If at this point r^* appears in $\mathcal{R}_L$ or $\mathcal{R}_R$, we are done, as this means that we can reconstruct all of $\lhd$, and, as a consequence, all of $\sqsubset$, so assume this is not the case. We will now describe another rule that can be used to extend $\lhd_L$ or $\lhd_R$.

(M3) Set $s = |C_M|$, and consider an arbitrary vote r in $\mathcal{R}_M$. Suppose that some candidate from $C_L \cup C_R$ appears in top s positions in r; let c be the highest-ranked such candidate. Note that if $c \notin \{c^L, c^R\}$, we can stop and output "no": as every good order $\lhd$ has $\lhd_L$ as its prefix and $\lhd_R$ as its suffix, no extension of r is single-peaked with respect to $\lhd$.

Now, suppose that $c = c^L$, and c appears in position i in r. Observe that $i \neq 1$, as otherwise (L2) would be applicable. Let C' be the set of candidates that appear above c^L in r; we have $C' \subseteq C_M$. Also, set $C'' = C_M \setminus C'$. Note that $1 \leq |C'| < s$, so $C'' \neq \emptyset$. Consider a good order $\lhd$. We know that the set $C' \cup \{c^L\}$ is contiguous with respect to $\lhd$; this implies that $C'' \cup \{c^R\}$ is also contiguous with respect to $\lhd$. However, to append C' to $\lhd_L$ or to prepend C'' to $\lhd_R$, we have to decide in which order the elements of these sets should be added. To this end, we use r^*: if c_m, who is the top candidate in r^*, is not in C', we append the elements of C' to $\lhd_L$, in reverse order of their appearance in r^*, and otherwise, i.e., if $c_m \notin C''$, we prepend the elements of C'' to $\lhd_R$, in order of their appearance in r^*. To see why this is correct, observe that, if $c_m \notin C'$, this means that in $\lhd$ candidate c_m appears to the right of all candidates in $C' \cup \{c^L\}$, and therefore any vote that ranks c_m first (such as r^*) lists the candidates in $C' \cup \{c^L\}$ in the right-to-left order with respect to $\lhd$; if $c_m \notin C''$, the argument is symmetric.

The analysis above is for the case $c = c^L$; the case $c = c^R$ can be analyzed in the same way.

If an application of (M3) extends $\lhd_L$ or $\lhd_R$ in a non-trivial way, some of the rules (L2)–(L3), (R2)–(R3), and, subsequently, (M2) may become applicable, and then (M3) may be applicable again. We apply these rules in an arbitrary order, with one constraint: we only invoke (M3) when other rules are not applicable. This is because this rule assumes that the candidates in $C_L \cup C_R$ form a suffix of r^*. When none of the rules applies, we have a configuration that enjoys the following properties:

- All votes in $\mathcal{R}_L$ only rank candidates in C_L, and all votes in $\mathcal{R}_R$ only rank candidates in C_R.
- For each candidate c that appears in $\lhd_L$ we have $\mathcal{R}(c) \subseteq \mathcal{R}_L$, and the order of voters in $\mathcal{R}(c)$ is given by $\sqsubset_L$; similarly, for each candidate c' that appears in $\lhd_R$, we have $\mathcal{R}(c') \subseteq \mathcal{R}_R$, and the order of voters in $\mathcal{R}(c')$ is given by $\sqsubset_R$.
- No vote in $\mathcal{R}_M$ contains a candidate from $C_L \cup C_R$ in top $|C_M|$ positions.

Now, let us focus on the votes in $\mathcal{R}_M \setminus \{r^*\}$. Suppose first that no candidate from $C_L \cup C_R$ appears in these votes. (We treat this case separately to build up reader's intuition; however, analysis for this case, which is presented below, is subsumed by the analysis for the general case.) In this case, we can call our algorithm recursively on the instance that consists of candidates in C_M and votes in $\mathcal{R}_M \cup \{r^*|_{C_M}\}$. If this turns out to be a "no"-instance, then our input instance is a "no"-instance as well. On the other hand, suppose that the recursive call produces a solution $\widehat{\mathcal{R}^M}$; let r' and r'' be the first and the last vote in $\widehat{\mathcal{R}^M}$ (these are total orders over C_M). We then order the votes in $\mathcal{R}_M$ in accordance with $\widehat{\mathcal{R}^M}$, and obtain a total order of all votes in $\mathcal{R}$ by prepending $\sqsubset_L$ and appending $\sqsubset_R$ to this ordering. Denote the resulting order by $\sqsubset$. We will argue that $\mathcal{R}$ is possibly single-crossing with respect to $\sqsubset$, by explicitly showing how to complete each order in $\mathcal{R}$. (We could argue that $\mathcal{R}$ is seemingly single-crossing with respect to that order and invoke Theorem 4, but it is useful to have an explicit description of the solution.) Specifically, we proceed as follows.

We use Theorem 4 to extend the votes in $\mathcal{R}_L$ and $\mathcal{R}_R$ to total orders over C_L and C_R, respectively; recall that, since these are top orders and rules (L2) and (R2) ensure that they are seemingly single-crossing with respect to $\sqsubset_L$ and $\sqsubset_R$, respectively, this is always possible. Observe that the rightmost voter in $\mathcal{R}_L$ orders the candidates in C_L according to the reverse of $\lhd_L$, and the leftmost voter in $\mathcal{R}_R$ orders the candidates in C_R according to $\lhd_R$. Next, we complete all votes as follows.

- to each vote in $\mathcal{R}_L$ we append r' followed by $r^*|_{C_R}$ (by construction, $r^*|_{C_R}$ coincides with $\lhd_R$);
- to each vote in $\mathcal{R}_M \setminus \{r^*\}$ we append $r^*|_{C_L \cup C_R}$;
- to each vote in $\mathcal{R}_R$ we append r'' followed by $r^*|_{C_L}$ (by construction $r^*|_{C_L}$ is the reverse of $\lhd_L$).

It is immediate that this profile is single-crossing as long as the one returned by the recursive call is.

The situation is a bit more complicated if some voters in $\mathcal{R}_M$ rank some of the candidates in $C_L \cup C_R$, as this imposes additional restrictions on how these votes can be ordered. Note, however, that if none of these votes disagrees with r^* on $C_L \cup C_R$ (in the sense that for every vote $r \in \mathcal{R}_M$ and every pair of candidates $a, b \in C_L \cup C_R$ such that $a \succ_{r^*} b$ either a and b are both unranked in r or $a \succ_r b$), the procedure described in the previous paragraph still applies.

Thus, suppose that some votes in $\mathcal{R}_M$ disagree with r^* on $C_L \cup C_R$. Observe that every such vote ranks all candidates in C_M: indeed, as (M3) does not apply, candidates in $C_L \cup C_R$ cannot appear in top $|C_M|$ positions in r, so, they appear in position $|C_M| + 1$ and lower, whereas the top $|C_M|$ positions are taken up by candidates in C_M. The following simple observation will be useful in our analysis.

Lemma 11. *Suppose that two votes $r, r' \in \mathcal{R}_M$ disagree with each other on $C_L \cup C_M$, i.e., there exist $c, c' \in C_L \cup C_R$ such that $c \succ_r c'$, but $c \succ_{r'} c'$. Then if $c, c' \in C_L$ or $c, c' \in C_R$, we have a "no"-instance of our problem. Further, if $c \in C_L$, $c' \in C_R$ then $r \sqsubset r'$ in any good order $\sqsubset$, and, if $c \in C_R$, $c' \in C_L$ then $r' \sqsubset r$ in any good order $\sqsubset$.*

Proof. If r and r' appear in $\mathcal{R}_M$, this means that for every good axis $\lhd$ it holds that the top candidates of r and r' appear after all candidates in C_L. Therefore, both r and r' should rank the candidates in C_L according to the reverse on $\lhd_L$, so in particular they cannot disagree on C_L. By the same argument, they cannot disagree on C_R. Now, if $c \in C_L$, $c' \in C_R$, the votes in $\mathcal{R}(c)$ precede both r and r' in any good order $\sqsubset$, so if we place r' before r, candidates c and c' would cross more than once. For the case $c \in C_R$, $c' \in C_L$, the argument is similar. $\square$

We will now modify our algorithm for the case where C_M and $C_L \cup C_R$ are cleanly separated, as follows.

First, we invoke Lemma 11 for every pair of votes in $\mathcal{R}_M$ that disagree on $C_L \cup C_R$ and output "no" if it tells us that we have a "no"-instance of our problem. If this does not happen, Lemma 11 produces a relation on votes in $\mathcal{R}_M$; if this relation is not a partial order, i.e., has cycles, we output "no" as well. Thus, suppose that this relation is indeed a partial order. Let $\mathcal{R}'_M = \{r \mid r'|_{C_M} = r \text{ for some } r' \in \mathcal{R}_M\}$; the elements of $\mathcal{R}'_M$ are top orders on C_M. We call our algorithm recursively on $(C_M, \mathcal{R}'_M)$, and return "no" if it returns "no". Now, suppose our recursive call returns a solution, i.e., an ordering $\sqsubset'$ of $\mathcal{R}'_M$. and a completion of partial orders in $\mathcal{R}'_M$ to total orders over C_M that is single-crossing with respect to $\sqsubset'$. We now construct a total order $\sqsubset$ on $\mathcal{R}$ as follows.

(Flip) If there exist two orders $r^-, r^+ \in \mathcal{R}_M$ that disagree both on C_M and on $C_L \cup C_R$ and Lemma 11 implies that $r^- \sqsubset r^+$ in every good order $\sqsubset$, but $r^+|_{C_M} \sqsubset' r^-|_{C_M}$, we flip $\sqsubset'$, i.e., construct an order $\sqsubset''$ such that for every $r^1, r^2 \in \mathcal{R}'_M$ it holds that $r^1 \sqsubset'' r^2$ if and only if $r^2 \sqsubset' r^1$,and then set $\sqsubset' := \sqsubset''$. Note that $\mathcal{R}'_M$ remains single-crossing with respect to $\sqsubset'$ after the flip. We only apply this operation once.

(Order-M) If $r, r' \in \mathcal{R}_M$, $r|_{C_M} \neq r'|_{C_M}$, and $r|_{C_M} \sqsubset' r'|_{C_M}$, we set $r \sqsubset r'$. If $r, r' \in \mathcal{R}_M$ and $r|_{C_M} = r'|_{C_M}$, since $\mathcal{R}$ contains no duplicates, r and r' differ on some candidates in $C_L \cup C_R$, in which case we use Lemma 11 to order them. This allows us to order all of $\mathcal{R}_M$.

(Order-LR) We set $r \sqsubset r'$ whenever $r, r' \in \mathcal{R}_L$ and $r \sqsubset_L r'$ or $r, r' \in \mathcal{R}_R$ and $r \sqsubset_R r'$. Also, we set $r \sqsubset r'$ whenever $r \in \mathcal{R}_L$, $r' \notin \mathcal{R}_L$ or $r' \in \mathcal{R}_R$, $r \notin \mathcal{R}_R$.

This procedure produces an ordering of the votes in $\mathcal{R}$; if $\mathcal{R}$ is seemingly single-crossing with respect to this ordering, we use Theorem 4 to complete $\mathcal{R}$.

Throughout the execution of the algorithm, we output "no" only if there is direct evidence that the input profile cannot be extended to a complete profile that is single-crossing. It remains to argue that if our procedure outputs an order $\sqsubset$, but $\mathcal{R}$ is not seemingly single-crossing with respect to $\sqsubset$, then we have a "no"-instance of our problem. Note that a violation of the single-crossing property may only be caused by voters in $\mathcal{R}_M$ ordering candidates in $C_L \cup C_R$. Any such violation takes the following form: $r, r' \in \mathcal{R}_M$, $r \sqsubset r'$, but Lemma 11 says that r' should precede r in every good order of votes, i.e., $a \succ_r b$, $b \succ_{r'} a$, where $a \in C_R$, $b \in C_L$. Further, this can only happen if $r|_{C_M}$ and r'_{C_M} are both total orders over C_M, and, moreover, r and r' disagree on both C_M and $C_L \cup C_R$ (if r and r' agree on C_M, (Order-M) orders them according to Lemma 11). In particular, this means that the two orders r^- and r^+ mentioned in (Flip) are well-defined.

Suppose for the sake of contradiction that there is a good order of votes $\sqsubset^*$ for $\mathcal{R}$. Consider the set $\mathcal{R}^* = \{r|_{C_M}, r'|_{C_M}, r^-|_{C_M}, r^+|_{C_M}\}$; it contains at least two distinct elements, and all of its elements are total orders over C_M. If $\mathcal{R}^*$ is not single-crossing, we obviously have a "no"-instance of our problem, so assume that it is single-crossing. By Proposition 1 in [19], there are exactly two orders on $\mathcal{R}^*$ such that $\mathcal{R}^*$ is single-crossing with respect to these orders, and these orders are reverse of each other (we emphasize that $\mathcal{R}^*$ is a set, i.e., contains no duplicates). Only one of these orders places $r^-|_{C_M}$ before $r^+|_{C_M}$ (recall that $r^-|_{C_M}$ and $r^+|_{C_M}$ are distinct); denote this order by $\sqsubset^1$.

As $\sqsubset^*$ is good, it has to agree with $\sqsubset^1$, i.e., for all $r^1, r^2 \in \{r, r', r^-, r^+\}$ such that $r^1 \sqsubset^* r^2$ and $r^1|_{C_M} \neq r^2|_{C_M}$ it holds that $r^1|_{C_M} \sqsubset^1 r^2|_{C_M}$. Now, recall that our recursive call produced an ordering $\sqsubset'$ of $\mathcal{R}'_M$ such that $\mathcal{R}'_M$ is single-crossing with respect to $\sqsubset'$; moreover, after (Flip) has been executed, we have $r^-|_{C_M} \sqsubset' r^+|_{C_M}$. It follows that $\sqsubset'$ coincides with $\sqsubset^1$ on $\mathcal{R}^*$. During (Order-M) we placed r before r'; as we have $r|_{C_M} \neq r'|_{C_M}$, this means that $r|_{C_M} \sqsubset' r'|_{C_M}$, or, equivalently, $r|_{C_M} \sqsubset^1 r'|_{C_M}$. As $\sqsubset^*$ agrees with $\sqsubset^1$, we have $r \sqsubset^* r'$. But this is a contradiction with the fact that $a \succ_r b$, $b \succ_{r'} a$ for some $a \in C_R$, $b \in C_L$. Thus, there is no good order of the votes in $\mathcal{R}$, and, consequently, $\mathcal{R}$ cannot be extended to a single-crossing profile. □

5. Relaxing the Single-Crossing Condition

Throughout this paper, we implicitly assumed that voters' true preferences are total orders, and the reasons why voters submit partial orders have to do with computation and/or communication constraints. Alternatively, one can imagine that some voters are truly indifferent between certain candidates. It is not clear whether requiring the given profile of partial orders to extend to a single-crossing profile of total orders is the right generalization of the single-crossing condition to such settings. In fact, one can argue that in case of true indifferences seemingly single-crossing profiles are exactly the profiles that should be considered single-crossing: indeed, in such profiles no pair of alternatives can be observed to cross more than once. If we view being seemingly single-crossing as a desirable property of a profile in its own right, it is natural to ask whether it can be detected efficiently. However, this question turns out to be computationally difficult, even if we restrict ourselves to weak orders.

Proposition 12. *The problem of deciding if a profile of weak orders is seemingly single-crossing is* NP*-complete, even if we restrict the votes to have antichains of size at most 2.*

Proof. This statement follows immediately from Theorem 7 and Corollary 6. □

Now, in a seemingly single-crossing profile, as we progress from left to right, for a given pair of candidates a, b we may go from a voter who is indifferent between a and b to one who clearly prefers a to b and then to one who is indifferent between a and b again. It is perhaps more intuitive to require instead that the only allowable transitions are from $a \succ b$ to indifference between a and b to $b \succ a$, or vice versa. We will call such profiles *blockwise single-crossing*, since we move from a block of voters that prefer a to b to a block of voters that is indifferent to a block that prefers b to a. We remark that the *order restriction* as defined by Rothstein [40]—which corresponds to a definition of single-peakedness for weak orders—is exactly the blockwise single-crossing restriction for weak orders.

Definition 4. *A profile $\mathcal{R} = (r_1, \dots, r_n)$ of partial orders over a candidate set C is* blockwise single-crossing with respect to a total order $\sqsubset$ over $[n]$ *if for every pair of candidates $a, b \in C$ there exist indices $0 \leq k \leq \ell \leq n+1$ such that either (i) for all $1 \leq i \leq k$ we have $a \succ_{r_i} b$, for all $k < i < \ell$ candidates a and b are incomparable in r_i, and for all $\ell \leq i \leq n$ we have $b \succ_{r_i} a$, or, alternatively, (ii) for all $1 \leq i \leq k$ we have $b \succ_{r_i} a$, for all $k < i < \ell$ candidates a and b are incomparable in r_i, and for all $\ell \leq i \leq n$ we have $a \succ_{r_i} b$. A profile $\mathcal{R} = (r_1, \dots, r_n)$ of partial orders is* blockwise single-crossing *if it is blockwise single-crossing with respect to some total order $\sqsubset$ over $[n]$.*

Observe that profile $\mathcal{R}$ from Example 1 is not blockwise single-crossing with respect to $1 \sqsubset 2 \sqsubset 3 \sqsubset 4$: we go from $a \succ c$ to $a \perp_{\succ} c$ to $a \succ c$. Consequently, the profile $\mathcal{R}'$ from that example is not blockwise single-crossing.

Clearly, it is easy to check if a given profile of partial orders $\mathcal{R}$ is blockwise single-crossing with respect to a given order $\sqsubset$. Interestingly, while checking whether $\mathcal{R}$ is blockwise single-crossing (for any arbitrary order) appears to be more difficult, this problem turns out to be polynomial-time solvable as well.

Theorem 13. *There is a polynomial-time algorithm that given a profile of partial orders $\mathcal{R}$ checks whether $\mathcal{R}$ is blockwise single-crossing, and, if the answer is positive, outputs an ordering of the voters that witnesses this.*

Proof. We reduce our problem to the well-known consecutive-ones problem [9]. Recall that an instance of the consecutive-ones problem is given by a 0-1 matrix; the goal is to reorder the columns of this matrix so that in each row all the 1s appear consecutively. This problem is known to admit a linear-time algorithm [9, 28]. Reductions to consecutive-ones problem have been employed to devise efficient algorithms for checking whether a given profile of total orders is single-peaked [3] or single-crossing [11]; recently, the former result has been extended to weak orders [27]. Our reduction is similar to that of Bredereck et al. [11].

Given a profile of partial orders $\mathcal{R} = (r_1, \ldots, r_n)$ over a candidate set C, $|C| = m$, we construct a 0-1 matrix M as follows. We have one column for every voter in $\mathcal{R}$ and two rows for each ordered pair of distinct candidates (a, b), which we denote by $R^{a,b,+}$ and $R^{a,b,-}$. We place a 1 in the i-th column of $R^{a,b,+}$ if $a \succ_{r_i} b$ and we place 0 in this column otherwise. Also, we place a 0 in the i-th column of $R^{a,b,-}$ if $a \succ_{r_i} b$ and we place 1 in this column otherwise. This completes the reduction.

To see why this reduction is correct, suppose first that the input profile is blockwise single-crossing with respect to an order $\sqsubset$ on $[n]$; assume without loss of generality that this order is $1 \sqsubset \cdots \sqsubset n$. Consider an arbitrary pair of candidates $a, b \in C$. Swapping a and b if necessary, we can assume that there exist indices $0 \leq k \leq \ell \leq n+1$ such that for all $1 \leq i \leq k$ we have $a \succ_{r_i} b$, for all $k < i < \ell$ candidates a and b are incomparable in r_i, and for all $\ell \leq i \leq n$ we have $b \succ_{r_i} a$. Then in $R^{a,b,+}$ we have 1s up to position k, followed by 0s, in $R^{a,b,-}$ we have 0s up to position k, followed by 1s, in $R^{b,a,+}$ we have 0s up to position $\ell - 1$, followed by 1s, and in $R^{b,a,-}$ we have 1s up to position $\ell - 1$, followed by 0s. Thus, ordering the rows of M according to $\sqsubset$ results in all 1s appearing consecutively in each row.

Conversely, suppose that we managed to reorder the columns of M so that all 1s appear consecutively in each row; assume without loss of generality that the matrix M itself has this property. We will argue that in this case $\mathcal{R}$ is blockwise single-crossing with respect to $1 \sqsubset \dots \sqsubset n$. Consider an arbitrary pair of candidates $a, b \in C$; we will show that for this pair the blockwise single-crossing property is satisfied. Since 1s appear consecutively in $R^{a,b,+}$, we know that voters who prefer a to b form a contiguous segment of $\sqsubset$. Similarly, since 1s appear consecutively in $R^{b,a,+}$, we know that voters who prefer b to a form a contiguous segment of $\sqsubset$. Thus, we can assume without loss of generality that all voters who prefer a to b appear before all voters who prefer b to a in $\sqsubset$. Hence, $\sqsubset$ is of the form $S_1 \sqsubset A \sqsubset S_2 \sqsubset B \sqsubset S_3$, where A is the set of voters who prefer a to b, b is the set of voters who prefer b to a, and the sets S_1, S_2, and S_3 consist of voters who are indifferent between a and b (given two set of voters X, Y, we write $X \sqsubset Y$ to indicate that in $\sqsubset$ all voters from X precede all voters from Y). We consider the following cases:

- $A = \emptyset$, $B = \emptyset$. Then all voters are indifferent between a and b, and the blockwise single-crossing property for this pair is trivially satisfied.

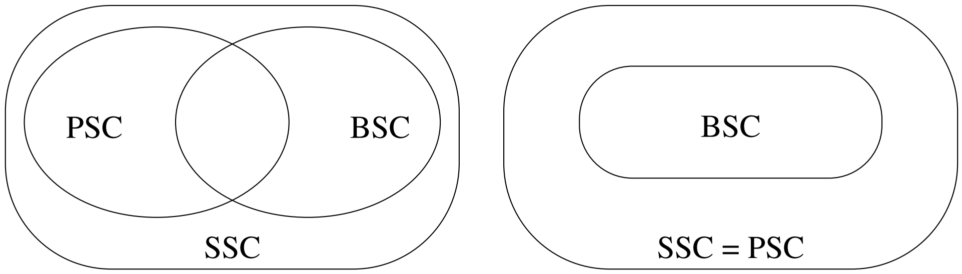
- $A = \emptyset$, $B \neq \emptyset$. We will argue that in this case either $S_3 = \emptyset$ or $S_1 \cup S_2 = \emptyset$, and hence the pair a, b satisfies the blockwise single-crossing property. Indeed, if this is not the case, in row $R^{b,a,-}$, we have 1s in positions corresponding to voters from $S_1 \cup S_2$ and S_3 and 0s in positions corresponding to voters in B, so a 0 appears between two 1s.

- $A \neq \emptyset$, $B = \emptyset$. This case is symmetric to the previous case: by considering the row $R^{a,b,-}$, we can conclude that either $S_1 = \emptyset$ or $S_2 \cup S_3 = \emptyset$.

- $A \neq \emptyset$, $B \neq \emptyset$. If $S_1 \neq \emptyset$, then we have a 0 between two 1s in row $R^{a,b,-}$ (1s correspond to elements of S_1 and B, 0 corresponds to an element of A). Similarly, if $S_3 \neq \emptyset$, then we have a 0 between two 1s in row $R^{b,a,-}$ (1s correspond to elements of A and S_3, 0 corresponds to an element of B). Thus, we have $S_1 = S_3 = \emptyset$ in this case.

In all four cases the pair a, b does not provide a witness that $\mathcal{R}$ is not blockwise single-crossing with respect to $\sqsubset$. As a and b were chosen arbitrarily, this completes the proof. □

Finally, let us consider the relation between possibly single-crossing, seemingly single-crossing, and blockwise single-crossing. The following theorem makes the containment relation between the corresponding classes of partial and weak orders precise; for an overview we refer the reader to Figure 1.



(a) partial orders (b) weak orders

Figure 1: The relation of possibly single-crossing (PSC), seemingly single-crossing (SSC) and blockwise single-crossing (BSC) in partial orders and weak orders.

Theorem 14. *Fix a set of candidates C, a set of voters $[n]$, and the order $1 \sqsubset \cdots \sqsubset n$. Let PSC, BSC, and SSC be the set of profiles of partial orders that are possibly single-crossing, blockwise single-crossing, and seemingly single-crossing, respectively, with respect to the given order $\sqsubset$. Furthermore, let T be the set of profiles of total orders and W the set of profiles of weak orders. Then the following statements hold.*

1. $PSC \subset SSC$
2. $BSC \subset SSC$
3. $BSC \not\subset PSC$ *and* $PSC \not\subset BSC$
4. $W \cap BSC \subset W \cap PSC = W \cap SSC$
5. $T \cap BSC = T \cap PSC = T \cap SSC$

Proof. Statement 1 follows from the fact that if a profile of partial orders $\mathcal{R}$ can be extended to a single-crossing profile of total orders, then $\mathcal{R}$ necessarily has to satisfy the seemingly single-crossing property. The containment $PSC \subset SSC$ is strict because of Example 1.

Statement 2 follows immediately from the definitions. To see that the containment is strict consider the following profile of three partial orders: $r_1 : a \perp_{\succ} b$, $r_2 : a \succ b$, and $r_3 : b \succ a$. This profile is seemingly single-crossing but not blockwise single-crossing with respect to $1 \sqsubset 2 \sqsubset 3$.

Let us consider Statement 3: $PSC \not\subset BSC$ follows from the counterexample in the previous paragraph. To show that $BSC \not\subset PSC$ requires a more sophisticated example, which is displayed in Figure 2. Towards a contradiction assume that $(\widehat{r}_1, \widehat{r}_2, \widehat{r}_3)$ is an extension of (r_1, r_2, r_3) to a single-crossing profile of total orders. Note that either $\widehat{r}_3 : c'' \succ a''$ or $\widehat{r}_3 : c' \succ a'$. Since $r_1 : c'' \succ a''$ or

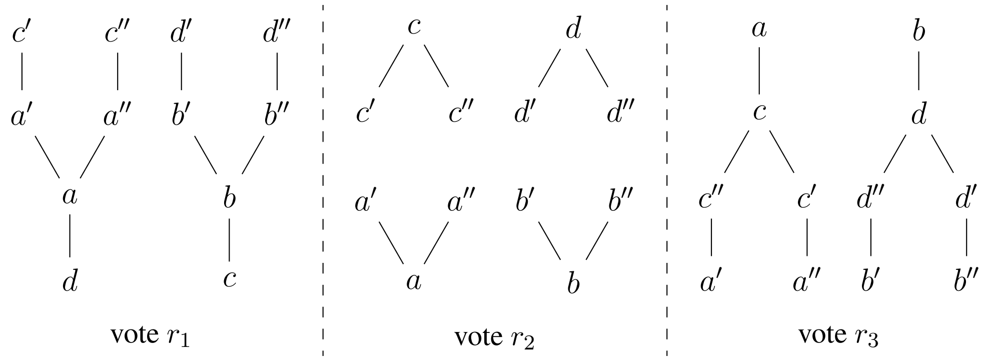


Figure 2: A profile of partial orders that is blockwise single-crossing but not possibly single-crossing.

$r_1 : c' \succ a'$, either $\widehat{r}_2 : c'' \succ a''$ or $\widehat{r}_2 : c' \succ a'$ has to hold and hence $\widehat{r}_2 : c \succ a$. Analogously, either $\widehat{r}_3 : d'' \succ b''$ or $\widehat{r}_3 : d' \succ b'$, from which we can conclude that $\widehat{r}_2 : d \succ b$. Since $r_3 : a \succ c$ and $\widehat{r}_2 : c \succ a$, the single-crossing property implies that $\widehat{r}_1 : c \succ a$. Further, since $r_3 : b \succ d$ and $\widehat{r}_2 : d \succ b$, the single-crossing property implies that $\widehat{r}_1 : d \succ b$. This implies that $r_1 : d \succ b \succ c \succ a \succ d$, a contradiction.

In Statement 4, $W \cap BSC \subset W \cap SSC$ follows immediately from the definitions and the counterexample of Statement 2. The equality $W \cap PSC = W \cap SSC$ is Corollary 6. To see Statement 5, note that the respective definitions do not differ for total orders. □

6. Conclusions and Open Problems

We summarize our results for SCC and SCC-F in Table 1. It is instructive to compare them with recent results of Lackner [30] and Fitzsimmons [27] for single-peaked preferences. Lackner proves that one can check in polynomial time whether a profile of partial votes is single-peaked with respect to a given axis. In contrast, verifying the possibly single-crossing property appears to be hard even if the order of the votes is fixed, though we have not been able to obtain a formal hardness result. Moreover, powerful algorithmic techniques that are very useful for working with incomplete single-peaked preferences, such as reductions to 2-SAT and to the consecutive ones problem, while applicable, appear to produce much weaker results in our setting. These are indications that incomplete single-crossing preferences are more difficult to work with than incomplete single-peaked preferences, and new insights are required.

orders	SCC-F	SCC
partial	open FPT(a, b) (Thm. 3)	NPc (Thm. 7)
weak	P (Thm. 4)	NPc (Thm. 7) FPT(k) (Thm. 9)
top	P (Thm. 4)	P under additional restr. (Thm. 10)

Table 1: Complexity results: P stands for "polynomial-time solvable", NPc stands for "NP-complete", FPT(x) stands for "fixed-parameter tractable with respect to parameter x".

The computational complexity of some of our problems remains open. Perhaps the most intriguing is the complexity of TO-SCC (top orders, arbitrary order of votes) and PO-SCC-F (partial orders, fixed order of votes). Also, given that much of the real-life election data consists of incomplete preference orders, it would be interesting to check how often real-life elections admit single-crossing extensions.

Acknowledgements.

Piotr Faliszewski was supported in part by the NCN grants DEC-2012/06/M/ST1/00358, and DEC-2011/03/B/ST6/01393, and by the AGH University grant 11.11.230.124 (statutory research). Martin Lackner was supported by the European Research Council (ERC) under grant number 639945 (ACCORD). Svetlana Obraztsova was supported by European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) – Research Funding Program: THALES, by RFFI 14-01-00156-a grant, and by a STSM grant from the COST Action IC120.

Bibliography

[1] K. Arrow. *Social Choice and Individual Values*. John Wiley and Sons, 1951.

[2] B. Aspvall, M. F. Plass, and R. E. Tarjan. A linear-time algorithm for testing the truth of certain quantified Boolean formulas. *Information Processing Letters*, 8(3):121–123, 1979.

[3] J. Bartholdi, III and M. Trick. Stable matching with preferences derived from a psychological model. *Operations Research Letters*, 5(4):165–169, 1986.

[4] D. Baumeister and J. Rothe. Taking the final step to a full dichotomy of the possible winner problem in pure scoring rules. *Information Processing Letters*, 112(5):186–190, 2012.

[5] D. Baumeister, P. Faliszewski, J. Lang, and J. Rothe. Campaigns for lazy voters: Truncated ballots. In *Proceedings of AAMAS-2012*, pages 577–584, June 2012.

[6] N. Betzler and B. Dorn. Towards a dichotomy of finding possible winners in elections based on scoring rules. *Journal of Computer and System Sciences*, 76(8):812–836, 2010.

[7] N. Betzler, A. Slinko, and J. Uhlmann. On the computation of fully proportional representation. *Journal of Artificial Intelligence Research*, 47:475–519, 2013.

[8] D. Black. *The Theory of Committees and Elections*. Cambridge University Press, 1958.

[9] K. Booth and G. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *Journal of Computer and System Sciences*, 13(3):335–379, 1976.

[10] F. Brandt, M. Brill, E. Hemaspaandra, and L. A. Hemaspaandra. Bypassing combinatorial protections: Polynomial-time algorithms for single-peaked electorates. *Journal of Artificial Intelligence Research*, 53:439–496, 2015.

[11] R. Bredereck, J. Chen, and G. Woeginger. A characterization of the single-crossing domain. *Social Choice and Welfare*, 41(4):989–998, 2013.

[12] R. Bredereck, J. Chen, and G. J. Woeginger. Are there any nicely structured preference profiles nearby? *Mathematical Social Sciences*, 79:61–73, 2016.

[13] B. Chamberlin and P. Courant. Representative deliberations and representative decisions: Proportional representation and the Borda rule. *American Political Science Review*, 77(3):718–733, 1983.

[14] V. Conitzer. Eliciting single-peaked preferences using comparison queries. *Journal of Artificial Intelligence Research*, 35:161–191, 2009.

[15] D. Cornaz, L. Galand, and O. Spanjaard. Bounded single-peaked width and proportional representation. In *Proceedings of ECAI-2012*, pages 270–275, August 2012.

[16] D. Cornaz, L. Galand, and O. Spanjaard. Kemeny elections with bounded single-peaked or single-crossing width. In *Proceedings of IJCAI-2013*, pages 76–82, 2013.

[17] M. Cygan, F. V. Fomin, Ł. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized algorithms*. Springer, 2015.

[18] R. G. Downey and M. R. Fellows. *Fundamentals of parameterized complexity*. Springer, 2013.

[19] E. Elkind and P. Faliszewski. Recognizing 1-Euclidean preferences: An alternative approach. In *Proceedings of SAGT-2014*, pages 146–157, 2014.

[20] E. Elkind and M. Lackner. Structure in dichotomous preferences. In *Proceedings of IJCAI-2015*, pages 2019–2025, 2015.

[21] E. Elkind, P. Faliszewski, and A. Slinko. Clone structures in voters' preferences. In *Proceedings of EC-2012*, pages 496–513, 2012.

[22] E. Elkind, P. Faliszewski, and P. Skowron. A characterization of the single-peaked single-crossing domain. In *Proceedings of AAAI-2014*, pages 654–660, 2014.

[23] E. Elkind, M. Lackner, and D. Peters. Structured preferences. In U. Endriss, editor, *Trends in Computational Social Choice*. AI Access, 2017.

[24] G. Erdélyi, M. Lackner, and A. Pfandler. Computational aspects of nearly single-peaked electorates. *Journal of Artificial Intelligence Research*, 58: 297–337, 2017.

[25] B. Escoffier, J. Lang, and M. Öztürk. Single-peaked consistency and its complexity. In *Proceedings of ECAI-2008*, pages 366–370, July 2008.

[26] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra. The complexity of manipulative attacks in nearly single-peaked electorates. *Artificial Intelligence*, 207:69–99, 2014.

[27] Zack Fitzsimmons. Single-peaked consistency for weak orders is easy. In *Proceedings of TARK-2015*, pages 103–110, 2015.

[28] Michel Habib, Ross McConnell, Christophe Paul, and Laurent Viennot. Lex-bfs and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing. *Theoretical Computer Science*, 234(1):59–84, 2000.

[29] K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *Proceedings of MPREF-2005*, pages 124–129, 2005.

[30] M. Lackner. Incomplete preferences in single-peaked electorates. In *Proceedings of AAAI-2014*, pages 742–748, 2014.

[31] T. Lu and C. Boutilier. Budgeted social choice: From consensus to personalized decision making. In *Proceedings of IJCAI-2011*, pages 280–286, 2011.

[32] K. Magiera and P. Faliszewski. How hard is control in single-crossing elections? In *Proceedings of ECAI-2014*, pages 579–584, 2014.

[33] N. Mattei and T. Walsh. Preflib: A library for preferences. In *Proceedings of ADT-2013*, pages 259–270, 2013.

[34] J. Mirrlees. An exploration in the theory of optimal income taxation. *Review of Economic Studies*, 38:175–208, 1971.

[35] H. Moulin. *Axioms of Cooperative Decision Making*. Cambridge University Press, 1991.

[36] N. Narodytska and T. Walsh. The computational impact of partial votes on strategic voting. In *Proceedings of ECAI-2014*, pages 657–662, 2014.

[37] J. Opatrny. Total ordering problem. *SIAM Journal on Computing*, 8(1): 111–114, 1979.

[38] A. Procaccia, J. Rosenschein, and A. Zohar. On the complexity of achieving proportional representation. *Social Choice and Welfare*, 30(3):353–362, 2008.

[39] K. W. S. Roberts. Voting over income tax schedules. *Journal of Public Economics*, 8(3):329–340, 1977.

[40] P. Rothstein. Order restricted preferences and majority rule. *Social choice and Welfare*, 7(4):331–342, 1990.

[41] P. Skowron, L. Yu, P. Faliszewski, and E. Elkind. The complexity of fully proportional representation for single-crossing electorates. *Theoretical Computer Science*, 569:43–57, 2015.

[42] L. Xia and V. Conitzer. Determining possible and necessary winners given partial orders. *Journal of Artificial Intelligence Research*, 41:25–67, 2011.