The Complexity of Safe Manipulation under Scoring Rules

Egor Ianovski* **Lan Yu†** **Edith Elkind†** **Mark C. Wilson***
* Department of Computer Science, University of Auckland
† Division of Mathematical Sciences, Nanyang Technological University, Singapore

Abstract

[Slinko and White, 2008] have recently introduced a new model of coalitional manipulation of voting rules under limited communication, which they call *safe strategic voting*. The computational aspects of this model were first studied by [Hazon and Elkind, 2010], who provide polynomial-time algorithms for finding a safe strategic vote under k-approval and the Bucklin rule. In this paper, we answer an open question of [Hazon and Elkind, 2010] by presenting a polynomial-time algorithm for finding a safe strategic vote under the Borda rule. Our results for Borda generalize to several interesting classes of scoring rules.

1 Introduction

Voting is an important tool for preference aggregation in multi-agent systems [Ephrati and Rosenschein, 1997]. However, essentially all voting rules are vulnerable to *manipulation*, i.e., voters may have an incentive to misrepresent their preferences to get a more desirable outcome [Gibbard, 1973; Satterthwaite, 1975]. This problem becomes especially severe when we assume that manipulating voters can form coalitions in order to coordinate their actions.

To assess the potential impact of coalitional manipulation in the context of multi-agent systems, where both the number of voters and the number of alternatives can be large, one needs to know whether manipulators can easily compute their optimal strategies. For this reason, the algorithmic complexity of coalitional manipulation received considerable attention in the multiagent research community (see [Faliszewski *et al.*, 2008; 2010; Xia *et al.*, 2009; 2010; Zuckerman *et al.*, 2009]). Indeed, the complexity of coalitional manipulation under the Borda rule is commonly viewed as one of the most interesting open questions in the emerging area of computational social choice.

Now, standard models of manipulation presuppose perfect communication and co-ordination within the manipulating coalition. Such conditions would be difficult—if not impossible—to attain in reasonably large elections where there could be thousands of agents opting to manipulate. In light of this, [Slinko and White, 2008] have proposed a more plausible model, where the coalition consists of voters with identical preferences, a single coalition member broadcasts a strategic vote, and every voter in the coalition has a choice between casting the proposed strategic vote or voting sincerely.

Since the original manipulator does not know how many followers he will have, manipulation in this sense can be risky: it may happen that, while the manipulation is useful when the number of followers is small, it may result in an undesirable candidate being elected when many coalition members opt to manipulate, or vice versa. Thus, the manipulator may want to identify a strategic vote that never produces an undesirable outcome, no matter how many coalition members choose to follow him. [Slinko and White, 2008] call such a vote a *safe strategic vote*. The main result of [Slinko and White, 2008] is that the Gibbard–Satterthwaite theorem can be extended to this notion of manipulation: every manipulable voting rule is safely manipulable. The probability of safe manipulation has been recently investigated by [Reyhani and Wilson, 2010].

Just as in the standard model of coalitional manipulation, the practical significance of this form of manipulation depends on whether a safe strategic vote can be computed efficiently. This problem was first studied by [Hazon and Elkind, 2010], who provided polynomial-time algorithms for finding a safe strategic vote, even with weighted voters, under the k-approval rule (for fixed k which may depend on the number of candidates — this includes the plurality and veto rules) and the Bucklin rule. For Borda, they have shown a **coNP**-hardness result for weighted voters; however, the unweighted case was left as an open question.

In this paper, we continue the investigation of algorithmic complexity of safe strategic voting initiated by [Hazon and Elkind, 2010]. Our main result is a polynomial-time algorithm for finding a safe strategic vote under the Borda rule with unweighted voters. The Borda rule is perhaps the most prominent representative of a large class of voting rules known as *scoring rules*: recall that a scoring rule for an m-candidate election is described by a vector $(\alpha_1, \dots, \alpha_m)$ with $\alpha_1 \geq \dots \geq \alpha_m$, where each candidate receives α_i points from each voter that ranks him in position i. Thus, it is natural to ask whether our algorithmic results for Borda extend to other scoring rules. We answer this question in the affirmative, by identifying a large class of scoring rules for which we can find a safe strategic vote in polynomial time. In particular, this is the case for all *top-heavy* scoring rules, i.e., rules that satisfy $\alpha_1 - \alpha_2 \geq \alpha_i - \alpha_{i+1}$ for all $i = 2, \ldots, m-1$, as well as for all rules where the increments $\alpha_i - \alpha_{i+1}$ take a small number of different values.

Another interesting class of scoring rules that is not captured by our original approach consists of rules for which the scoring vector consists of a constant number of "blocks". For such rules, we propose a polynomial-time algorithm for finding safe strategic votes that is based on a different idea. Though we have not (yet!) designed polynomial-time algorithms for safe manipulation under *all* scoring rules, we believe that our results contribute to the understanding of the computational complexity of safe strategic voting.

2 Preliminaries

An *election* $E = (A, V)$ is described by a set of *alternatives* (sometimes called *candidates*) A, $|A| = m$, and a set of *voters* V, $|V| = n$. Every voter $i \in V$ is associated with a *preference ordering* R_i, which is a total order over A; we will also refer to a voter's preference ordering as his *type*. For instance, if $A = \{a, b, c, d\}$, a voter may rank c first, followed by b, followed by d, followed by a; we will abbreviate the type of this voter as $cbda$. The list of all voters' preference orderings $\mathcal{R} = (R_1, \ldots, R_n)$ is called the *preference profile*. A *voting correspondence* is a mapping $\mathcal{F}$ from the set of all preference profiles over A to the set $2^A \setminus \{\emptyset\}$ of all non-empty subsets of A; the elements of the set $\mathcal{F}(\mathcal{R})$ are called the *election winners*. A voting correspondence $\mathcal{F}$ is called a *voting rule* if there is a unique winner for each preference profile, i.e., $|\mathcal{F}(\mathcal{R})| = 1$ for any preference profile $\mathcal{R}$. A voting correspondence is said to be *anonymous* if its output does not change when the entries of $\mathcal{R}$ are permuted.

A *scoring rule* $\mathcal{F}_\alpha$ is a voting correspondence given by a score vector $\alpha = (\alpha_1, ..., \alpha_m)$ that satisfies $\alpha_1 \geq ... \geq \alpha_m$. Under $\mathcal{F}_\alpha$, each candidate a receives $\sum_i \alpha_i s_i$ points, where s_i is the number of voters ranking a in position i. The candidate(s) with the highest score are the election winners. The *k-approval rule* is the scoring rule with $\alpha_i = 1$ for $i \leq k$, $\alpha_i = 0$ for $i > k$; here k is a given function of m. The *Borda rule* is the scoring rule with the score vector $(m-1, m-2, \ldots, 0)$. Note that a scoring rule is defined for a fixed number of candidates m. Since we are interested in asymptotic complexity results, we will abuse notation and use the term "scoring rule" to refer to *efficiently computable families of scoring rules*, i.e., sets that contain a scoring rule for each value of m and admit an efficient algorithm that, given a value of m, computes the scoring rule corresponding to m. Note that both Borda and k-approval can be viewed as efficiently computable families of scoring rules.

The definitions in the previous paragraph assume that all voters have equal power. A more general setup allows for *weighted* voters. In a weighted election, we are given a vector $\mathbf{w} = (w_1, \ldots, w_n)$, where w_i is the weight of the i-th voter; all weights w_i are required to be positive integers given in binary. To determine the winner of a weighted election under a voting rule $\mathcal{F}$, we replace a voter with weight w_i with w_i unweighted voters and apply $\mathcal{F}$ to the resulting profile.

To transform a voting correspondence into a voting rule. we need a *tie-breaking rule*, i.e., a mapping $T : 2^A \to A$ that given a set of tied alternatives $S \subseteq A$ outputs a single alternative $T(S)$ such that $T(S) \in S$. Clearly, if $\mathcal{F}$ is a voting correspondence, then the composition of T and $\mathcal{F}$, i.e., the mapping $T \circ \mathcal{F}$ given by $(T \circ \mathcal{F})(\mathcal{R}) = T(\mathcal{F}(\mathcal{R}))$, is a voting rule. A tie-breaking rule T is said to be *lexicographic* if there exists an ordering $\succ$ over A such that, for each set $S \subseteq A$, $T(S)$ is the first element in S according to $\succ$.

Consider an election $E = (A, V)$ where the voters' preferences are given by a profile $\mathcal{R}$, and a voting rule $\mathcal{F}$. Fix a voter v of type R, and let M denote the set of all voters in V who also have type R. We are interested in situations where v announces a manipulative vote $L \neq R$ and a subset of voters in M may decide to vote L, while all the remaining voters vote truthfully. Given a set of voters X, we denote by $\mathcal{R}_{-X}(L)$ the preference profile obtained from $\mathcal{R}$ by replacing the vote of every voter in X with L. Let $w = \mathcal{F}(\mathcal{R})$ be the election winner if everyone votes truthfully. We will say that an alternative $a \in A$ is *good* if it is ranked above w in R; we say that a is *bad* if it is ranked below w in R. We denote the set of all good alternatives by G, and the set of all bad alternative by B.

The following definition is adapted from [Slinko and White, 2008; Hazon and Elkind, 2010].

Definition 1. *Consider an election $E = (A, V)$ with a preference profile $\mathcal{R}$, a voting rule $\mathcal{F}$, a type $R \in \mathcal{R}$, and the set $M = \{i \in V \mid R_i = R\}$. Let L be a preference order over A. Then*

- *L is* safe *for M if for any $X \subseteq M$ we have $\mathcal{F}(\mathcal{R}_{-X}(L)) \in G \cup \{w\}$.*
- *L is* strategic *for M if there exists an $X \subseteq M$ and a $g \in G$ such that $\mathcal{F}(\mathcal{R}_{-X}(L)) = g$.*

A vote is said to be a safe strategic vote *if it is both safe and strategic.*

Example 1. Consider a Borda election with $A = \{a, b, c, d\}$, 5 voters of type $bacd$, 4 voters of type $abcd$, 4 voters of type $dcab$, 2 voters of type $badc$, 2 voters of type $cdba$, 1 voter of type $bdac$, 1 voter of type $dacb$ and 1 voter of type $dcba$. Under truthful voting, b is first with 35 points, a is second with 33, c and d are tied for last with 26. Suppose that a voter of type $abcd$ opts to manipulate, so $G = \{a\}$, $B = \{c, d\}$. Consider first the vote $adcb$. This vote is strategic because if the manipulating coalition is of size 2, b's score will fall to 31, below that of a, while d's score will only rise to 30, so the highest scoring alternative is a, which is in G. However, the vote is not safe: if the manipulating coalition is of size 4, then d will score 34 points and thus the election winner will be an alternative in B.

On the other hand, consider the vote $acbd$. This time, even if all voters of type $abcd$ manipulate, c's score will only rise to 30, while b's will fall to 31, and the election winner will be a. Thus, the vote $abcd$ is a safe strategic vote.

We will now define the algorithmic problem that is the focus of this work.

EXISTSAFE($\mathcal{F}$): given an election $E = (A, V)$ with a preference profile $\mathcal{R}$, a type R, and the set M that consists of all voters of type R in $\mathcal{R}$, does M have a safe strategic vote under a voting rule $\mathcal{F}$?

3 Complexity of ExistSafe for Borda

In this section, we study the complexity of ExistSafe with unweighted voters under the Borda rule. Almost all of our preliminary results (Lemmas 1–5 and Propositions 1 and 2) hold for all scoring rules, so we state and prove them for an arbitrary fixed scoring rule $\mathcal{F}$. We then present our main result of this section—a polynomial-time algorithm for ExistSafe(Borda). In the next section, we show how to extend our proof to a large class of scoring rules.

To simplify the presentation, throughout this section, we assume that ties are broken adversarially to the manipulator, i.e., the tie-breaking rule is lexicographic with respect to the ordering obtained by reversing the manipulator's preference ordering. However, all of our proofs can be adapted to work for arbitrary lexicographic tie-breaking rules; we will comment on this after presenting the main proof.

Throughout this section, we assume that a scoring rule $\mathcal{F}$ is fixed, and we are given an election (A, V) with a preference profile $\mathcal{R}$ and a manipulating set $M \subseteq V$ that consists of all voters with a certain type R. We set $w = \mathcal{F}(\mathcal{R})$, and let G and B be the candidates ranked above and below w in R, respectively. We will refer to the position of w in R as w's *sincere* position.

For any $X \subseteq M$, any $a \in A$ and any vote L, we denote by $S_X(a, L)$ the score of a in $\mathcal{F}(\mathcal{R}_{-X}(L))$; when $X = \emptyset$, the quantity $S_X(a, L)$ does not depend on L, so we omit L and write $S_\emptyset(a)$. Note that the score $S_X(a, L)$ is well-defined even if L is a *partial* vote, i.e., if the positions of some alternatives in L are unknown, as long as the position of a itself is known. Let $\mathcal{L}_k$ denote the set of all votes that rank w in position k. We say that an alternative $a \in A$ *overtakes* $b \in A$ if a loses to b under truthful voting, but beats it when all voters in M vote L. A position k is said to be *promising* if there exists an $L \in \mathcal{L}_k$ such that $S_X(w, L) > S_X(b, L)$ for all $b \in B$ and all $X \subseteq M$. Observe that if k is a promising position, then there exists a safe vote with w ranked in position k. Further, w's sincere position is necessarily promising.

We start by establishing the existence of an interval of w's positions where strategic votes exist, and an interval where safe votes exist. We then show that if these intervals intersect, we are assured that either the lowest promising position or the highest non-promising position fall into that intersection.

Lemma 1. *If there exists a safe strategic vote, then there exists a safe strategic vote in which every good alternative is ranked above every bad alternative.*

Proof. Assume we have a safe strategic vote L with some $b_1 \in B$ ranked above some $g_1 \in G$. Construct L' by swapping b_1 with g_1. This does not decrease the score of g_1, does not increase the score of b_1, and does not change the score of any other alternative.

Consider a subset $X \subseteq M$. Since L is safe, the winner at $\mathcal{R}_{-X}(L)$ is either w or some $g \in G$. Since $S_X(a, L') \geq S_X(a, L)$ for all $a \in G \cup \{w\}$ and $S_X(b, L') \leq S_X(b, L)$ for all $b \in B$, the winner at $\mathcal{R}_{-X}(L')$ is also either w or some $g' \in G$, so vote L' is also safe.

Further, since L is strategic, there is a subset $Y \subseteq M$ such that the winner at $\mathcal{R}_{-Y}(L)$ is some $g \in G$. Since $S_Y(a, L') \geq S_Y(a, L)$ for all $a \in G$ and $S_Y(b, L') \leq S_Y(b, L)$ for all $b \in B \cup \{w\}$, the winner at $\mathcal{R}_{-Y}(L')$ is also some $g' \in G$. Thus, L' is also strategic. □

Given this result, from now on we will assume that every vote we deal with has the good alternatives ranked above the bad alternatives.

Lemma 2. *If k is a promising position, $k-1$ is a promising position.*

Proof. Since k is a promising position, there is a vote $L \in \mathcal{L}_k$ such that $S_X(w, L) > S_X(b, L)$ for all $X \subseteq M$ and all $b \in B$. Construct L' from L by swapping w with the alternative directly above it. This does not decrease the score of w, and does not increase the score of any $b \in B$. Hence, we have $S_X(w, L') > S_X(b, L')$ for all $X \subseteq M$ and all $b \in B$. i.e., L' is a witness that $k-1$ is a promising position. □

Lemma 3. *If k is a promising position and there is no strategic vote in $\mathcal{L}_k$, there is no strategic vote in $\mathcal{L}_{k-1}$.*

Proof. We prove the contrapositive. Suppose there is a strategic vote $L \in \mathcal{L}_{k-1}$, i.e., there exists a set $X \subseteq M$ and a candidate $g \in G$ such that $S_X(g, L) > S_X(a, L)$ for all $a \in B \cup \{w\}$. Consider the vote L' obtained from L by swapping w with the alternative directly below it. This does not increase the score of w and does not decrease the score of g, so we have $S_X(g, L') > S_X(w, L')$. Further, since k is a promising position, we can reorder the bad alternatives in L' (without changing the positions of w and g) so that the resulting vote L'' satisfies $S_Y(w, L'') > S_Y(b, L'')$ for all $Y \subseteq M$ and all $b \in B$. Thus, substituting $X = Y$, we have $S_X(g, L'') = S_X(g, L') > S_X(w, L') = S_X(w, L'') > S_X(b, L'')$ for all $b \in B$, so the vote L'' is also strategic. □

Lemma 4. *If there is no safe vote in $\mathcal{L}_k$, there is no safe vote in $\mathcal{L}_{k+1}$.*

Proof. We prove the contrapositive. Assume there exists a safe vote $L \in \mathcal{L}_{k+1}$. Consider the alternative a ranked directly above w in L. If $a \in G$, w must be ranked at or above its sincere position, since by Lemma 1 every good alternative is ranked above every bad alternative. Since w's sincere position is promising, Lemma 2 implies that k is a promising position as well, so there must be a safe vote in $\mathcal{L}_k$.

On the other hand, if $a \in B$, construct $L' \in \mathcal{L}_k$ by swapping w with a. This swap does not increase the score of any $b \in B$ and does not decrease the score of w or any $g \in G$. Therefore, if $L \in \mathcal{L}_{k+1}$ is safe, so is $L' \in \mathcal{L}_k$. □

Lemma 5. *If k is not a promising position, and some $L \in \mathcal{L}_k$ is safe, L is also strategic.*

Proof. Since k is not promising, there exists some manipulating coalition $X \subseteq M$ such that for some $b \in B$ it holds that $S_X(b, L) > S_X(w, L)$ and hence w is not the winner at $\mathcal{R}_{-X}(L)$. Since L is safe, the winner at $\mathcal{R}_{-X}(L)$ must be some candidate $g \in G$. Therefore L is strategic. □

Lemma 6. *If k is a promising position, and $\mathcal{L}_k$ contains a vote L such that $S_X(g, L) > S_X(w, L)$ for some $g \in G$ and $X \subseteq M$, then $\mathcal{L}_k$ contains a safe strategic vote.*

Proof. Since k is a promising position, we can reorder the bad alternatives in L without changing the positions of g and w so that the resulting vote L' satisfies $S_Y(w, L') > S_Y(b, L')$ for any $b \in B$ and any $Y \subseteq M$, i.e., L' is safe. Further, substituting $X = Y$, we obtain $S_X(g, L') = S_X(g, L) > S_X(w, L) = S_X(w, L') > S_X(b, L')$ for any $b \in B$, so L' is also strategic. $\square$

Proposition 1. *Let k be the lowest promising position. If a safe strategic vote exists, then there exists one in $\mathcal{L}_k \cup \mathcal{L}_{k+1}$.*

Proof. We prove the contrapositive. Assume that there are no safe strategic votes in $\mathcal{L}_k \cup \mathcal{L}_{k+1}$. By Lemma 6, $\mathcal{L}_k$ cannot contain a strategic vote, and hence by Lemma 3, there are no strategic votes in $\mathcal{L}_i$ for any $i \leq k$.

Further, by Lemma 5, any safe vote in $\mathcal{L}_{k+1}$ is strategic. Hence, there can be no safe vote in $\mathcal{L}_{k+1}$, so by Lemma 4, there are no safe votes in $\mathcal{L}_j$ for any $j \geq k+1$. Thus, we have argued that if there are no safe strategic votes in $\mathcal{L}_k \cup L_{k+1}$, there are no safe strategic votes in $\mathcal{L}_j$ for any value of j. $\square$

The next lemma shows that, when considering the effects of a manipulation on a given pair of candidates, it is enough to check what happens when all voters in M choose to cast a manipulative vote.

Lemma 7. *For any vote L and any candidates $x, y \in A$ such that $S_\emptyset(x) \leq S_\emptyset(y)$ we have $S_M(x, L) > S_M(y, L)$ if and only if $S_X(x, L) > S_X(y, L)$ for some $X \subseteq M$.*

Proof. The "only if" direction is obvious: we can take $X = M$. To prove the "if" direction, suppose that $S_X(x, L) > S_X(y, L)$ for some $X \subseteq M$. Let γ and δ denote, respectively, the change in x's and y's score when one voter in M switches from R to L; we have $S_Z(x, L) = S_\emptyset(x, L) + \gamma|Z|$, $S_Z(y, L) = S_\emptyset(y, L) + \delta|Z|$ for any $Z \subseteq M$. Since we have $S_\emptyset(x, L) \leq S_\emptyset(y, L)$, it follows that $\gamma > \delta$. Therefore, $S_M(x, L) = S_X(x, L) + \gamma(|M| - |X|) > S_X(y, L) + \delta(|M| - |X|) = S_M(y, L)$. $\square$

Proposition 2. *Testing whether a given position k is promising is in* $\mathbf{P}$.

Proof. We know that w's sincere position is promising, so, by Lemma 2, if $k \leq |G|$ we can output "yes". If $k > |G|$, we construct a (partial) vote L as follows. We rank w in position k, and place the good alternatives in the top $|G|$ positions in any order. Then we construct a bipartite graph (U, D), where $U = B$ and the set D consists of positions in L unoccupied by the alternatives in $G \cup \{w\}$. There is an edge from $b \in B$ to a position $p \in D$ if the partial vote L' that ranks b in position p satisfies $S_M(w, L') > S_M(b, L')$. Using Lemma 7, it is easy to check that k is a promising position if and only if (U, D) contains a perfect matching (which can be checked in polynomial time). $\square$

The following crucial lemma does not extend to general scoring rules if $m \geq 5$ (counterexample omitted for space reasons).

Lemma 8. *Under the Borda rule, if k is a promising position and there is no strategic vote in $\mathcal{L}_k$, then for any vote $L \in \mathcal{L}_{k+1}$ there is at most one $g \in G$ such that $S_M(g, L) > S_M(w, L)$.*

Proof. Assume, for contradiction, that there is no strategic vote in $\mathcal{L}_k$, but there exists a vote $L \in \mathcal{L}_{k+1}$ and two good alternatives $g_1, g_2 \in G$ such that $S_M(g, L) > S_M(w, L)$ for $g \in \{g_1, g_2\}$. Since only one of g_1 and g_2 can be ranked in the top position in L, assume without loss of generality that g_2 is ranked second or lower. Construct L' from L by swapping g_2 with the alternative right above it and then swapping w with the alternative right above it. Note that since $S_M(g_2, L) > S_M(w, L)$, g_2 is ranked above w in L, and therefore the latter swap does not affect the position of g_2. We have $L' \in \mathcal{L}_k$; further, we have $S_M(w, L') = S_M(w, L) + |M|$ and $S_M(g_2, L') = S_M(g_2, L) + |M|$, because both w and g_2 gain one point for every voter that opts to rank them one position higher. Thus, we have $S_M(g_2, L') = S_M(g_2, L) + |M| > S_M(w, L) + |M| = S_M(w, L')$. Since k is a promising position, by Lemma 6 we can transform L' into a safe strategic vote in $\mathcal{L}_k$, a contradiction. $\square$

Theorem 1. EXISTSAFE(Borda) *is in* $\mathbf{P}$.

Proof. Given Proposition 1, we only need to check votes in $\mathcal{L}_k$ and $\mathcal{L}_{k+1}$, where k is the lowest promising position. First, we use the algorithm of Proposition 2 to find k. Then we try to construct a safe strategic vote in $\mathcal{L}_k$. To this end, for each $g \in G$ we construct a vote L^g that ranks g first, ranks w in position k and ranks all other candidates arbitrarily, and check whether $S_M(g, L^g) > S_M(w, L^g)$. If this is the case for some $g \in G$, since k is promising, by Lemma 6 we can transform L^g into a safe strategic vote. If we have $S_M(g, L^g) \leq S_M(w, L^g)$ for all $g \in G$, then no $g \in G$ can overtake w and hence no vote in $\mathcal{L}_k$ can be strategic.

Now, suppose that no strategic vote has been found in $\mathcal{L}_k$. Then, by Lemma 8, if there is a safe strategic vote $L \in \mathcal{L}_{k+1}$, then there is at most one alternative g such that $S_M(g, L) > S_M(w, L)$. By Lemma 7, this implies that for any $X \subseteq M$ and any $g' \in G \setminus \{g\}$ we have $S_X(g', L) \leq S_X(w, L)$. Since L is safe, this means that for any $X \subseteq M$ and any $b \in B$ it holds that $S_X(b, L) < \max\{S_X(w, L), S_X(g, L)\}$. Observe also that if we swap g with the alternative ranked in the top position in L (which, by Lemma 1, we can assume to be an alternative in G), the resulting vote will remain safe and strategic. Hence, if $\mathcal{L}_{k+1}$ contains a safe strategic vote, then there exists some alternative g and a vote $L \in \mathcal{L}_{k+1}$ such that L ranks g first and for any $b \in B$ and any $X \subseteq M$ we have $S_X(b, L) < \max\{S_X(w, L), S_X(g, L)\}$. Thus, to find a safe strategic vote in $\mathcal{L}_{k+1}$, we proceed as follows. For each $g \in G$ we construct a partial vote L^g that ranks g first, followed by other votes in G in an arbitrary order, and ranks w in position $k+1$ (note that since $k+1$ is not promising, the sincere position of w is k or higher, so $|G| \leq k-1$, i.e., after the alternatives in G are ranked, the position $k+1$ is still unoccupied).

Then, as in the proof of Proposition 2, we construct a bipartite graph (U, D), where $U = B$ and the set D consists of positions in L unoccupied by the alternatives in $G \cup \{w\}$.

There is an edge from $b \in B$ to a position $p \in D$ if the partial vote L' that ranks b in position p satisfies $S_X(b, L) < \max\{S_X(w, L), S_X(g, L)\}$ for all $X \subseteq M$. The latter condition can be verified in polynomial time, since the Borda rule is anonymous and hence it suffices to check this condition for an arbitrary manipulating coalition of size s for $s = 1, \ldots, |M|$. It is easy to see that $\mathcal{L}_{k+1}$ contains a safe strategic vote if and only if (U, D) contains a perfect matching. □

It remains to explain how to extend our algorithm to arbitrary lexicographic tie-breaking rules. This can be achieved by replacing each expression of the form "$S_X(a, L) > S_X(b, L)$" by "either (a) $S_X(a, L) > S_X(b, L)$ or (b) $S_X(a, L) = S_X(b, L)$ and $a \succ b$", where $\succ$ is the given lexicographic order. Equivalently, assuming that $\succ$ orders the candidates as $a_m \succ \ldots \succ a_1$, we can set $\varepsilon = 1/(m^2)$ and modify the score of each candidate by setting $S'_X(a_i, L) = S_X(a_i, L) + i\varepsilon$. Since we consider scoring rules with integer entries only, the modified scores of any two candidates are different, and $S'_X(a, L) > S'_X(b, L)$ if and only if (a) $S_X(a, L) > S_X(b, L)$ or (b) $S_X(a, L) = S_X(b, L)$ and $a \succ b$. We can now use the modified scores throughout the proof. This argument allows us to assume that ties never occur; the proofs in Section 4 make use of this assumption.

4 Other scoring rules

In this section, we describe polynomial-time algorithms for finding a safe manipulation for several classes of scoring rules. First, we present results that can be obtained by generalizing the arguments used for Borda. Then, we describe an algorithm for scoring rules with a constant number of different scores, which uses a different idea.

4.1 Extensions of Theorem 1

All results of Section 3 except Lemma 8 and Theorem 1 hold for all scoring rules. We will now show that we can prove an analogue of Lemma 8 for a large class of scoring rules, and, as a result, obtain polynomial-time algorithms for finding a safe strategic vote with respect to such rules.

We first introduce additional notation. Given a scoring rule $\mathcal{F}_\alpha$ with $\alpha = (\alpha_1, \ldots, \alpha_m)$, let $\Delta\alpha_i = \alpha_i - \alpha_{i+1}$ for $i = 1, \ldots, m-1$, and set $\mathcal{I}(\alpha) = \{\Delta\alpha_i \mid i = 1, \ldots, m-1\}$. Let $\Delta_{\min} = \min\{\Delta\alpha_i \mid i = 1, \ldots, m-1\}$, $\Delta_{\max} = \max\{\Delta\alpha_i \mid i = 1, \ldots, m-1\}$. Set $s(t) = \sum_{j=1}^{t} \Delta\alpha_j$, and define $r(\alpha) = \min\{t \mid s(t) \geq \Delta_{\max}\}$.

Let k be the lowest promising position, and suppose that there is no safe strategic vote in $\mathcal{L}_k$. Consider a vote $L \in \mathcal{L}_{k+1}$, and let $G^*(L) = \{g \in G \mid S_M(g, L) > S_M(w, L)\}$. By Lemma 7, for any alternative $g' \in G \setminus G^*(L)$ and any $X \subseteq M$ we have $S_X(g', L) \leq S_X(w, L)$, i.e., the alternatives in $G \setminus G^*(L)$ never win. Thus we can move the alternatives in $G^*(L)$ into top $|G^*(L)|$ positions (note that since k is the lowest promising position, we have $|G^*(L)| < k$): the resulting vote L' is safe and strategic as long as L is. Therefore, we can limit our attention to the votes $L \in \mathcal{L}_{k+1}$ that rank the alternatives in $G^*(L)$ in top $|G^*(L)|$ positions; we will say that any such vote is *nice*.

Lemma 9. *Let k be the lowest promising position, and suppose that there is no strategic vote in $\mathcal{L}_k$. Then any nice vote $L \in \mathcal{L}_{k+1}$ satisfies $|G^*(L)| \leq r(\alpha)$.*

Proof. Let $r = r(\alpha)$ and suppose for the sake of contradiction that $|G^*(L)| > r$. Since L is nice, some alternative g is ranked first in L, and some alternative g' is ranked in position $r + 1$ in L. We construct the vote L' by swapping g and g' in L, and then swapping w with the alternative right above it. The resulting vote L' is in $\mathcal{L}_k$. By our choice of r, we have $s(r) \geq \Delta\alpha_k$. Further, since $g' \in G^*(L)$, we get $S_M(g', L) > S_M(w, L)$. Therefore, we have

$$\begin{aligned} S_M(g', L') &= S_M(g', L) + s(r)|M| \\ &> S_M(w, L) + \Delta\alpha_k|M| = S_M(w, L'). \end{aligned}$$

Since k is a promising position, by Lemma 6 we can transform L' into a safe strategic vote in $\mathcal{L}_k$, a contradiction. □

Now, we can modify the algorithm given in the proof of Theorem 1 as follows. Just as in that proof, we find the lowest promising position k and check if there is a safe strategic vote in $\mathcal{L}_k$. If the answer is "no", we try to construct a safe strategic vote in $\mathcal{L}_{k+1}$. Let $r = \min\{r(\alpha), |G|\}$. By Lemma 9, we only need to decide which alternatives to put in the top r positions; for any placement of good candidates in top r positions, we rank the remaining good candidates right below them, place w in position $k + 1$, and then use the bipartite graph-based algorithm to rank the bad candidates, as described in the proof of Theorem 1. There are at most $\binom{m}{r} r! = O(m^r)$ ways to fill the top r positions. Thus, we obtain the following result.

Theorem 2. ExistSafe$(\mathcal{F}_\alpha)$ *is in* $\mathbf{P}$ *for any scoring rule* $\mathcal{F}_\alpha$ *with* $r(\alpha) = O(1)$.

The condition of Theorem 2 is somewhat technical. However, it covers many interesting classes of scoring rules and we now present some consequences. For example, we say that a scoring rule $\mathcal{F}_\alpha$ is *top-heavy* if $\Delta\alpha_1 \geq \Delta\alpha_j$ for all j; equivalently, $r(\alpha) = 1$. Every *convex* scoring rule (one where $\Delta\alpha_i \geq \Delta\alpha_j$ when $i \geq j$) is top-heavy.

Corollary 1. ExistSafe$(\mathcal{F}_\alpha)$ *is in* $\mathbf{P}$ *for any top-heavy scoring rule* $\mathcal{F}_\alpha$.

We can extend Corollary 1 to the case where the score vector consists of "blocks", with entries in each block being equal. Formally, we say that a scoring rule $\mathcal{F}_\alpha$ with the vector $\alpha = (\alpha_1 \ldots, \alpha_m)$ is *blockwise* with respect to a vector $\beta = (\beta_1, \ldots, \beta_\ell)$ if $\alpha = (\beta_1, \ldots, \beta_1, \beta_2, \ldots, \beta_2, \ldots, \beta_\ell, \ldots, \beta_\ell)$; with $\beta_1 > \ldots > \beta_\ell$. Let $\Delta\beta_j = \beta_j - \beta_{j+1}$ for $j = 1, \ldots, \ell - 1$. If $\mathcal{F}_\alpha$ is blockwise with respect to β, we say that it is *blockwise top-heavy* if $\Delta\beta_1 \geq \Delta\beta_j$ for all j.

Given a blockwise top-heavy rule $\mathcal{F}_\alpha$, let q be the size of the first block, i.e., $q = |\{\alpha_i \mid \alpha_i = \beta_1\}|$. We have, for each $i = 1, \ldots, m-1$, $s(q) = \sum_{j=1}^{q} \Delta\alpha_j = \alpha_{q+1} - \alpha_q = \Delta\beta_1 \geq \Delta\alpha_i$ and hence $r(\alpha) \leq q$. This yields the following result.

Corollary 2. ExistSafe$(\mathcal{F}_\alpha)$ *is in* $\mathbf{P}$ *for any blockwise top-heavy scoring rule* $\mathcal{F}_\alpha$ *with first block of constant size.*

Now, set $\rho(\alpha) = \lceil \frac{\Delta_{\max}}{\Delta_{\min}} \rceil$ if $\Delta_{\min} > 0$ and $\rho(\alpha) = +\infty$ otherwise. We have $\sum_{j=1}^{\rho(\alpha)} \Delta\alpha_j \geq \rho(\alpha)\Delta_{\min} \geq \Delta_{\max}$. Thus, $r(\alpha) \leq \rho(\alpha)$, and we obtain the following result.

Corollary 3. $\textsc{ExistSafe}(\mathcal{F}_\alpha)$ *is in* $\mathbf{P}$ *for any scoring rule* $\mathcal{F}_\alpha$ *such that* $\rho(\alpha) = O(1)$.

4.2 Scoring rules with a constant number of blocks

Let R be the manipulator's true preference ordering. For $j = 1, \dots, \ell$, let A_j be the set of candidates that obtain β_j points from R. Let $\hat{\ell} = \max\{j \mid A_j \cap G \neq \emptyset\}$. For $j = 1, \dots, \hat{\ell}$, let $g^*_{[j]}$ be the candidate in $A_j \cap G$ that has the highest score under truthful voting, i.e., $g^*_{[j]} = \arg\max_{a \in A_j \cap G} S_\emptyset(a)$ (recall that by the argument in the end of Section 3 we can assume that there are no ties). Set $G^* = \{g^*_{[j]} \mid j = 1, \dots, \hat{\ell}\}$; clearly, $|G^*| \leq \ell$. We will argue that to find a safe strategic vote with w in a non-promising position, it suffices to consider all possible placements of the candidates in G^* in top $|G^*|$ positions.

Lemma 10. *Consider two good alternatives* $g, g' \in G \cap A_j$ *and a safe strategic vote* L. *Let* L' *be the vote obtained from* L *by swapping* g *and* g'. *If* $S_\emptyset(g) \geq S_\emptyset(g')$ *and* g' *is ranked above* g *in* L, *then* L' *is also a safe strategic vote.*

Proof. Since L' ranks g higher than L does, we have $S_X(g, L') \geq S_X(g, L)$ for any $X \subseteq M$. Now, suppose L ranks g' in position i. Since $S_\emptyset(g) \geq S_\emptyset(g')$, we have

$$\begin{aligned} S_X(g, L') &= S_\emptyset(g) + (\alpha_i - \beta_j)|X| \\ &\geq S_\emptyset(g') + (\alpha_i - \beta_j)|X| = S_X(g', L). \end{aligned}$$

Thus, for any $X \subseteq M$, if some $a \in B \cup \{w\}$ is beaten by g or g' at $\mathcal{R}_{-X}(L)$, then a beaten by g at $\mathcal{R}_{-X}(L)$, i.e., L' is a safe strategic vote. □

By Lemma 10, when searching for a safe strategic vote, we may limit our attention to the votes that rank $g^*_{[j]}$ above all other alternatives in $A_j \cap G$, for all $j = 1, \dots, \hat{\ell}$. Moreover, for any such vote L we have $S_X(g^*_{[j]}, L) \geq S_X(g, L)$ for any $g \in G \cap A_j$ and any $X \subseteq M$, i.e., the ranking of alternatives in $G \setminus G^*$ is irrelevant to whether L is safe or not. Thus, given a safe strategic vote L, we can move the alternatives in G^* to the top $|G^*|$ positions; the resulting vote L' is also a safe strategic vote. Hence, when ranking the alternatives in G, it suffices to consider all possible assignments of the alternatives in G^* to the top $|G^*|$ positions, i.e., at most $\ell!$ possibilities. Now, substituting G^* for $G^*(L)$ in the proof of Theorem 2, we obtain the following result.

Theorem 3. $\textsc{ExistSafe}(\mathcal{F}_\alpha)$ *is in* $\mathbf{P}$ *for any scoring rule* $\mathcal{F}_\alpha$ *that is blockwise with respect to a vector* $(\beta_1, \dots, \beta_\ell)$ *with* $\ell = O(1)$.

5 Conclusions and Future Work

We have demonstrated that for many scoring rules, including the classic Borda rule, finding a safe strategic vote is easy. An obvious question is whether these results extend to all scoring rules (our proofs here do not), or whether safe manipulation is hard for some such rules. Other natural research directions include developing polynomial-time algorithms for ExistSafe under other common rules such as Copeland and Maximin.

In contrast with the results of [Hazon and Elkind, 2010], our results apply to unweighted voters only. It is tempting to conjecture that some of our proofs could be extended to weighted voters: for instance, the scoring rules considered in Theorem 3 can be viewed as a rather modest extension of k-approval and therefore one might expect that Theorem 3 holds for weighted voters as well. However, the proof of Theorem 8 in [Hazon and Elkind, 2010] suggests that this is unlikely: the construction in that proof essentially shows that both finding a safe strategic vote and checking if a given vote is safe is hard for the scoring rule given by the vector $(4, 3, 2, 1, 0, \dots, 0)$. Identifying scoring rules for which safe manipulation is easy even with weighted voters is an interesting problem as well.

References

[Ephrati and Rosenschein, 1997] E. Ephrati and J. Rosenschein. A heuristic technique for multi-agent planning. *Annals of Mathematics and Artificial Intelligence*, 20(1–4):13–67, 1997.

[Faliszewski *et al.*, 2008] P. Faliszewski, E. Hemaspaandra, and H. Schnoor. Copeland voting: Ties matter. In *AAMAS'08*, pages 983–990, 2008.

[Faliszewski *et al.*, 2010] P. Faliszewski, E. Hemaspaandra, and H. Schnoor. Manipulation of Copeland elections. In *AAMAS'10*, pages 367–374, 2010.

[Gibbard, 1973] A. Gibbard. Manipulation of voting schemes. *Econometrica*, 41(4):587–601, 1973.

[Hazon and Elkind, 2010] N. Hazon and E. Elkind. Complexity of safe strategic voting. In *SAGT'10*, pages 210–221, 2010.

[Reyhani and Wilson, 2010] R. Reyhani and M. C. Wilson. The probability of safe manipulation. In *COMSOC'10*, pages 423–430, 2010.

[Satterthwaite, 1975] M. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2):187–217, 1975.

[Slinko and White, 2008] A. Slinko and S. White. Non-dictatorial social choice rules are safely manipulable. In *COMSOC'08*, pages 403–413, 2008.

[Xia *et al.*, 2009] L. Xia, M. Zuckerman, V. Conitzer, A. Procaccia, and J. Rosenschein. Complexity of unweighted manipulation under some common voting rules. In *IJCAI'09*, pages 348–353, 2009.

[Xia *et al.*, 2010] L. Xia, V. Conitzer, and A. Procaccia. A scheduling approach to coalitional manipulation. In *ACM EC'10*, pages 275–284, 2010.

[Zuckerman *et al.*, 2009] M. Zuckerman, A. Procaccia, and J. Rosenschein. Algorithms for the coalitional manipulation problem. *Artificial Intelligence Journal*, 173(2):392–412, 2009.