On the Complexity of Borda Control in Single-Peaked Elections

Yongjie Yang
School of Information Science and Engineering, Central South University, Changsha, China
Chair of Economics, Saarland University, Saarbrücken, Germany
yyongjie@mmci.uni-saarland.de

ABSTRACT

Recent research reveals that many NP-hard voting problems in general become polynomial-time solvable in single-peaked elections. In contrast to these results, we prove for the first time that constructive control by adding/deleting votes for Borda are NP-hard even in single-peaked elections. On the other side, we prove that constructive control by adding/deleting votes/candidates for Borda are polynomial-time solvable in single-dived elections, which are elections obtained from single-peaked elections by reversing voters' preferences. Finally, we study constructive control by adding/deleting votes/candidates for Borda in single-peaked elections with k-truncated votes, i.e., each voter ranks only her top-k candidates, aiming at investigating how the values of k affect the complexity of these problems. For this purpose, we adopt the voting correspondences Borda$_{\uparrow}$, Borda$_{\downarrow}$ and Borda$_{av}$. We obtain many polynomial-time solvable results for k being a constant.

Keywords

Borda; control; single-peaked; parameterized complexity; social choice

1. INTRODUCTION

Voting is a common method for preference aggregation and collective decision-making, and has applications in multi-agent systems, political elections, web spam reduction, pattern recognition, etc. A major concern in voting is the potential existence of strategic individual who has an incentive to change the election results by controlling (reconstructing) the election, such as adding/deleting votes/candidates. There would be two goals that the strategic individual wants to reach: making a distinguished candidate p win the election, or making p lose the election. The former case is called a *constructive goal* and the latter case is called a *destructive goal*. We consider only control problems with a constructive goal in this paper. Fortunately, many control problems for commonly used voting correspondences are proved to be computationally hard to solve in general, even with the assumption that the strategic individual knows voters' preferences. As a result, the strategic individual may give up his plan to attack the election.

Nevertheless, in many real-world scenarios voters' preferences are subject to some combinatorial restrictions. One of the most well-studied restrictions is arguably the so-called single-peaked domain [7]. Generally, an election is single-peaked if there is an order of the candidates such that each voter's preference increases first and then decreases when the candidates are visited from one side to another side. See Figure 1 for an illustration. Single-peaked elections arise in many real-world scenarios. For instance, for candidates corresponding to numerical quantities, it is quite reasonable to assume that each voter has a particular favorite point in the range of candidates, and voters evaluate candidates by their proximity to this point. In addition, single-peaked elections have several nice properties. For instance, in single-peaked elections Condorcet cycles do not occur. A significant consequence is that one escapes from Arrow's impossibility theorem [1, 7]. Due to the importance of single-peaked elections, the complexity of strategic voting problems for many natural voting correspondences in single-peaked elections has been extensively studied recently. It turned out that many NP-hard problems become polynomial-time solvable in single-peaked elections [6, 8, 19].

In addition to combinatorial restrictions of the preferences of voters, in some real-world applications obtaining full rankings of the candidates is challenging. For instance, when the number of candidates is extremely large, it might be only possible to read the top-k ranked candidates in each vote. In this case, it is more efficient to ask voters to rank only their top-k candidates for some small integer k, though voters have complete preferences over all candidates. Votes with only the top-k candidates being ranked are referred to as k-truncated votes.

In this paper, we study the complexity of constructive control by adding/deleting votes/candidates for Borda in several specific settings, including single-peaked/dived elections and k-truncated single-peaked elections. Borda is arguably one of the most significant voting correspondences. In this setting, each voter gives points to the candidates according to her preference over the candidates. In particular, the i-th preferred candidate is given $m - i$ points, where m denotes the number of candidates. A Borda winner is a candidate with the highest total score. Problems related to Borda have been extensively studied in the literature [5, 10, 11, 25, 29, 30, 31, 32, 36]. In particular, it is known that constructive control by adding/deleting votes/canddiates for Borda in general elections are all NP-hard [9, 26, 31].

1.1 Our Contribution

Our main contributions are summarized as follows.

1. We prove that constructive control by adding/deleting votes for Borda are NP-hard even in single-peaked elections. These results stand in a sharp contrast to many of the previous polynomial-time solvability results in single-peaked elections. To the best of our knowledge, these are the first NP-hardness results

Appears in: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

of unweighted control problems for natural voting correspondences in single-peaked elections[1].

A line of research which received a considerable amount of attention very recently is to investigate the complexity of strategic voting problems in nearly single-peaked elections, such as t-swoop elections, t-additional axes single-peaked elections, elections with single-peaked width t, to name a few [15, 17, 33, 34, 35]. In general, each nearly single-peaked election is associated with an integer t, indicating the distance of the election to a single-peaked election. This line of research primarily aims at investigating the minimum value of t from which the complexity of specific strategic voting problems changes. Our NP-hardness results exactly pinpoint the complexity border of constructive control by adding/deleting votes for Borda in these nearly single-peaked elections.

2. We study control problems for Borda in single-dived elections. In particular, we prove that constructive control by adding/deleting votes/candidates for Borda are polynomial-time solvable in single-dived elections. To obtain these results, we derive a useful property of Borda scores of candidates in single-dived elections. This property may be useful for further studies of voting problems for Borda.

3. Finally, we consider k-truncated single-peaked elections, i.e., each voter ranks only her top-k preferred candidates and, moreover if all voters ranked all candidates we have a single-peaked election. We adopt the three extensions of Borda in elections with k-truncated votes, namely Borda$_\uparrow$, Borda$_\downarrow$ and Borda$_{av}$. These extensions have been studied in the literature (see, e.g., [4, 14, 28]). Let m be the number of candidates. In both Borda$_\uparrow$ and Borda$_{av}$, each candidate ranked in the i-th position, where $1 \leq i \leq k$, in a vote receives $m-i$ points. They differ in assigning the points to unranked candidates. In particular, in Borda$_\uparrow$ each unranked candidate receives 0 points, while in Borda$_{av}$ each unranked candidate receives $\frac{m-k-1}{2}$ points. In Borda$_\downarrow$, each candidate ranked in the i-th position, where $1 \leq i \leq k$, receives $k-i+1$ points, and each unranked candidate receives 0 points. We show that many control problems for Borda$_*$ where $* \in \{\uparrow, \downarrow, av\}$ are polynomial-time solvable when k is a constant. From parameterized complexity point of view, these problems are either shown to be fixed-parameter tractable (FPT) or lie in the class XP with respect to k.

Our main results are summarized in Tables 1 and 2.

1.2 Motivation

A motivation of the study of control problems is that the issues of adding/deleting votes/candidates occur in many electoral settings. For instance, in academic paper review process, it is up to the editor handling a paper to assign the paper to several reviewers, and before the final decision is made, the editor can add further reviewers. We refer to [18, 20, 23] for further concrete examples.

Using complexity as a barrier against strategic behavior has been suggested by many researchers. The key point is that if it is NP-hard for the strategic individual to successfully figure out how to reconstruct the election, he may refrain from attacking the voting. Therefore, whether a voting correspondence is resistant to control attacks has been recognized as a significant property to measure voting correspondences [2]. In addition, complexity analysis helps practitioners decide what kind of solution method is appropriate. For polynomial-time solvability results, we directly provide efficient algorithms. On the other hand, hardness results suggest that finding an exact solution is apt to be costly or impractical, and resorting to approximation or heuristic algorithms may be a necessary choice.

In addition, as Borda is a prevalent voting correspondence and single-peaked elections and elections with k-truncated votes are important, it makes sense to investigate the complexity of control problems for Borda in single-peaked elections and elections with k-truncated votes.

1.3 Preliminaries

Election. An *election* $\mathcal{E}$ is a tuple $(\mathcal{C}, \Pi_\mathcal{V})$, where $\mathcal{C}$ is a set of *candidates* and $\Pi_\mathcal{V}$ a multiset of *votes* cast by a set of voters (each voter casts one vote). Every vote $\pi \in \Pi_\mathcal{V}$ is defined as a linear order (permutation) over $\mathcal{C}$. For a linear order π and an element c in π, $\pi(c)$ is the *position* of c in π, i.e., the number of elements ordered before c plus 1. In addition, for an integer x with $1 \leq x \leq y$ where y is the number of elements in π, $\pi[x]$ is the x-th element of π, i.e., the one in the x-th position. Each vote π cast by a voter indicates the voter's preference over the candidates. In particular, a candidate a is preferred to another candidate b if $\pi(a) < \pi(b)$. A *voting correspondence* is a function that maps each election $\mathcal{E}$ to a nonempty subset of candidates, the *winners*. If there is only one winner, we call it the *unique winner*; otherwise, we call them *co-winners*.

For a linear order $\mathbf{x} = (x_1, x_2, ..., x_i)$ over a set $\{x_1, ..., x_i\}$, $\overleftarrow{\mathbf{x}}$ is the reversal of $\mathbf{x}$, i.e., $\overleftarrow{\mathbf{x}} = (x_i, x_{i-1}, ..., x_1)$. For $A \subseteq \{x_1, ..., x_i\}$, $\mathbf{x} \setminus A$ is the linear order obtained from $\mathbf{x}$ by removing all elements in A, and $\mathbf{x}[A] = \mathbf{x} \setminus (\{x_1, ..., x_i\} \setminus A)$. For instance, for $\mathbf{x} = (2, 4, 1, 7, 0)$ and $A = \{2, 7\}$, $\mathbf{x} \setminus A = (4, 1, 0)$ and $\mathbf{x}[A] = (2, 7)$. For two linear orders $\mathbf{x} = (x_1, x_2, ..., x_i)$ and $\mathbf{y} = (y_1, y_2, ..., y_j)$, we denote by $(\mathbf{x}, \mathbf{y})$ the linear order $(x_1, x_2, ..., x_i, y_1, y_2, ..., y_j)$. For $C \subseteq \mathcal{C}$, $\Pi_\mathcal{V}^C$ is the multiset of votes obtained from $\Pi_\mathcal{V}$ by replacing every $\pi \in \Pi_\mathcal{V}$ with $\pi[C]$.

Borda. A *positional scoring correspondence* is characterized by a scoring vector $\langle s(1), s(2), ..., s(m) \rangle$ such that $s(i) \geq s(j)$ for every $1 \leq i < j \leq m$, where m is the number of candidates. Moreover, each vote π gives $s(\pi(c))$ points to every candidate c. The winners are the candidates with the highest total score. The *Borda correspondence* is a positional scoring correspondence with the scoring vector $\langle m, m-1, ..., 0 \rangle$.

Single-peaked/dived election. An election $(\mathcal{C}, \Pi_\mathcal{V})$ is *single-peaked* if there is a linear order $(c_{\alpha(1)}, c_{\alpha(2)}, ..., c_{\alpha(m)})$ of $\mathcal{C}$, the so-called *harmonious order*, such that for every $\pi \in \Pi_\mathcal{V}$ and $1 \leq x < y < z \leq m$, it holds that $\pi(c_{\alpha(x)}) < \pi(c_{\alpha(y)})$ implies $\pi(c_{\alpha(y)}) < \pi(c_{\alpha(z)})$. The top ranked candidate in π is referred to as the *peak* of π. See Figure 1 for an illustration. *Single-dived elections* are obtained from single-peaked elections by reversing all votes.

Elections with k-truncated votes. A *k-truncation* π' of a vote π is the sublinear order of π with the first k candidates. Formally, π' is a permutation over the candidates in $\{c \in \mathcal{C} \mid \pi(c) \leq k\}$ such that $\pi'(c) = \pi(c)$ for every $c \in \mathcal{C}$ with $\pi(c) \leq k$. A *k-truncated vote* is a k-truncation of some vote. To avoid confusion, we use *complete votes* to refer to votes over $\mathcal{C}$ hereinafter. A *k-truncated election* is an election with k-truncated votes. A *k-truncated single-peaked election* is obtained from a single-peaked

[1] Faliszewski et al. [19] obtained several NP-hardness results of voting problems in single-peaked elections. However, their results hold only for weighted elections. Our NP-hardness results are for unweighted elections.

election by replacing each vote with its k-truncation. See Figure 1 for an illustration.

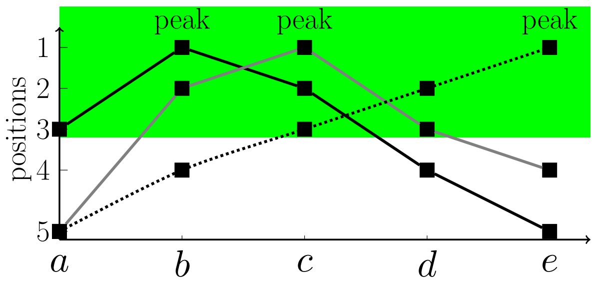

Figure 1: This figure illustrate a single-peaked election with 5 candidates a, b, c, d, e and 3 votes (b, c, a, d, e), (c, b, d, e, a) and (e, d, c, b, a), respectively. The 3-truncations (b, c, a), (c, b, d) and (e, d, c) of these votes are shown in the green area.

Borda with truncated votes. We mainly study three variants of the Borda correspondence in k-truncated elections, namely Borda$_\uparrow$, Borda$_\downarrow$ and Borda$_{av}$.

In Borda$_\uparrow$, each candidate in the i-th position, $1 \leq i \leq k$, in a vote receives $m - i$ points, and each unranked candidate receives 0 points. In Borda$_\downarrow$, each candidate in the i-th position, $1 \leq i \leq k$, receives $k - i + 1$ points, and each unranked candidate receives 0 points. In Borda$_{av}$, each candidate in the i-th position, $1 \leq i \leq k$, in a vote receives $m - i$ points, and each unranked candidate receives $\frac{m-k-1}{2}$ points. Or equivalently, each candidate in the i-th position receives $m + k - 2i + 1$ points, and every unranked candidate receives 0 points (multiply each score by 2 and then subtract $m - k - 1$ from it). In all three cases, candidates with the highest total score are the winners.

Problem formulations. We study four problems, namely CCAV, CCDV, CCAC and CCDC, where "CC" stand for "constructive control", and "AV|DV|AC|DC" stand for "adding votes|deleting votes|adding candidates|deleting candidates". The following formulations are based on k-truncated elections, but can be adapted to elections with complete votes naturally.

In the inputs of all problems, we have a k-truncated election $(\mathcal{C}, \Pi_\mathcal{V})$, a distinguished candidate $p \in \mathcal{C}$, and an integer $R \geq 0$. In CCDV, all candidates and k-truncated votes are registered. A *registered election* consists of all registered candidates and registered k-truncated votes. The question is whether p can win the registered election by deleting at most R many k-truncated votes. In CCAV, all candidates are registered, but not all k-truncated votes. In particular, a submultiset $\Pi_\mathcal{U} \subseteq \Pi_\mathcal{V}$ of unregistered k-truncated votes is given. The question is whether p can win the registered election by adding (registering) at most R many k-truncated votes in $\Pi_\mathcal{U}$.

By setting $k = |\mathcal{C}|$ in the definitions of CCAV and CCDV, we have the definitions of the problems in elections with complete votes.

In CCDC, all candidates and k-truncated votes are registered. The question is whether p can win the registered election by deleting at most R candidates from $\mathcal{C} \setminus \{p\}$. One may have observed that after the deletion of some candidates, there may be less than k candidates in a $\pi \in \Pi_\mathcal{V}$. So, we need to modify $\Pi_\mathcal{V}$ after the deletion of some candidates. As discussed previously, every voter has in fact a complete preference over all candidates. It is natural that when a voter observed that some of her top-k candidates were deleted, she will adjust her vote according to her preference. To capture this phenomena, we assume that each k-truncated vote $\pi \in \Pi_\mathcal{V}$ is associated with a complete vote $\bar{\pi}$ over $\mathcal{C}$ whose k-truncation is π. Moreover, if a subset C of candidates are deleted, each $\pi \in \Pi_\mathcal{V}$ will be reset as the t-truncation of $\bar{\pi} \setminus C$ where $t = \min\{k, |\mathcal{C} \setminus C|\}$. When $k = |\mathcal{C}|$, we have CCDC in elections with complete votes.

Now we discuss CCAC. In this case, all k-truncated votes are registered but not all candidates. In particular, a subset $\mathcal{D} \subseteq \mathcal{C} \setminus \{p\}$ of unregistered candidates such that $|\mathcal{C} \setminus \mathcal{D}| \geq k$ is given. The question is whether p can win the registered election by adding (registering) at most R candidates in $\mathcal{D}$. Analogous to CCDC, each k-truncated vote $\pi \in \Pi_\mathcal{V}$ is associated with a complete vote $\bar{\pi}$ such that π is the k-truncation of $\bar{\pi} \setminus \mathcal{D}$. Moreover, after adding a subset $C \subseteq \mathcal{D}$ of candidates, each $\pi \in \Pi_\mathcal{V}$ is reset as the k-truncation of $\bar{\pi} \setminus (\mathcal{D} \setminus C)$.

To define CCAC in elections with complete votes, we let $k = |\mathcal{C} \setminus \mathcal{D}|$. Moreover, after adding a subset $C \subseteq \mathcal{D}$ of candidates, a k-truncated vote $\pi \in \Pi_\mathcal{V}$ is reset as the $(k + |C|)$-truncation of $\bar{\pi} \setminus (\mathcal{D} \setminus C)$.

In all the above problems, we assume that the distinguished candidate p does not win in advance (i.e., p does not win before performing the corresponding manipulative action); since otherwise, we can immediately conclude that the instance is a YES-instance.

Following the convention, for each problem we distinguish between the *unique-winner model* and the *nonunique-winner model*. In the unique-winner model, winning an election means to be the unique winner, while in the nonunique-winner model, winning an election means to be the unique winner or to be a co-winner.

Our results hold for both the unique-winner and nonunique-winner models. We will not state this again in the theorems. We do not study destructive control by adding/deleting votes/candidates for Borda since they are polynomial-time solvable even in the general case [26, 31].

2. SINGLE-PEAKED AND SINGLE-DIVED ELECTIONS

In this section, we study control by adding/deleting votes/candidates for Borda in single-peaked and single-dived elections with complete votes. Our results in this section is summarized in Table 1. We first consider in single-peaked elections.

Recently, the complexity of control problems for many voting correspondences in single-peaked elections has been investigated. It turned out that many NP-hard control problems become polynomial-time solvable in single-peaked elections. In particular, CCAV and CCDV for all Condorcet-consistent voting correspondences as well as for several other non Condorcet-consistent voting correspondences such as Approval and r-Approval are polynomial-time solvable in single-peaked elections. We refer to [8, 19] for further details. However, whether these control problems for Borda in single-peaked elections are polynomial-time solvable remain open.

In this section, we show that CCAV and CCDV for Borda are NP-hard even in single-peaked elections. These are the first NP-hardness results of unweighted control problems for natural voting correspondences in single-peaked elections. In general elections, it is known that CCAV and CCDV for Borda are NP-hard [9, 31, 26]. However, all previous NP-hardness reductions do not apply to single-peaked elections since the elections constructed in these reductions are not single-peaked (as a matter of fact, these elections have very large distances from single-peaked elections).

THEOREM 1. *CCAV and CCDV for Borda are NP-hard even in single-peaked elections.*

	general case	single-peaked	single-dived
CCAV	NP-hard	**NP-hard**	**P**
CCDV	NP-hard	**NP-hard**	**P**
CCAC	NP-hard	?	**P**
CCDC	NP-hard	?	**P**

Table 1: Complexity of control problems for Borda. Our results are in bold. Entries filled with the "?" mean that the complexity of the corresponding problems remained open.

PROOF. We prove the theorem by reductions from the X3C problem defined as follows.

> Exact 3-Set Cover (X3C)
> *Input:* A universal set $U = \{c_1, c_2, ..., c_{3\kappa}\}$ and a collection $S = \{s_1, s_2, ..., s_n\}$ of 3-subsets of U.
> *Question:* Is there an $S' \subseteq S$ such that $|S'| = \kappa$ and each $c_i \in U$ appears in exactly one set of S'?

We assume that each $c_i \in U$ occurs in exactly 3 subsets of S. This assumption does not affect the NP-hardness [22]. Observe that under this assumption, $n = 3\kappa$.

Let $\mathcal{I} = (U = \{c_1, c_2, ..., c_{3\kappa}\}, S = \{s_1, s_2, ..., s_{3\kappa}\})$ be an instance of the X3C problem.

CCAV. Consider first the nonunique-winner model. We create an instance $\mathcal{E}_{\mathcal{I}} = (\mathcal{C}, \Pi_{\mathcal{V}}, \Pi_{\mathcal{U}}, p \in \mathcal{C}, R = \kappa)$ as follows.

Candidates $\mathcal{C}$**.** We create in total $6\kappa+1$ candidates. In particular, for each $c_x \in U$ we create two candidates c_x^L and c_x^R. In addition, we create a distinguished candidate p.

Let $\mathbf{a}$ be the linear order $(c_1^L, c_2^L, ..., c_{3\kappa}^L)$ and $\mathbf{b}$ the linear order $(c_1^R, c_2^R, ..., c_{3\kappa}^R)$. Moreover, let $\rhd = (\overleftarrow{\mathbf{a}}, p, \mathbf{b})$. We shall create votes that are single-peaked with respect to $\rhd$.

Registered Votes $\Pi_{\mathcal{V}} \setminus \Pi_{\mathcal{U}}$**.** We create in total $2\kappa + 2$ registered votes. In particular, we create $2\kappa + 1$ votes with peak at $c_{3\kappa}^R$ (i.e., each vote is defined as $(\overleftarrow{\mathbf{b}}, p, \mathbf{a})$. In addition, we have one vote defined as $(p, c_1^L, \mathbf{b}, \mathbf{a} \setminus \{c_1^L\})$.

Unregistered votes $\Pi_{\mathcal{U}}$**.** We create in total 3κ unregistered votes. In particular, for each $s_i \in S$, we create a vote π_i such that (1) $\pi_i(p) = 1$; and (2) for every $c_x \in U$, if $c_x \in s_i$ then $\pi_i(c_x^R) = 2x$ and $\pi_i(c_x^L) = 2x + 1$; otherwise, $\pi_i(c_x^L) = 2x$ and $\pi_i(c_x^R) = 2x + 1$.

It is fairly easy to check that all votes created above are single-peaked with respect to $\rhd$.

Finally, we set $R = \kappa$. Now we prove the correctness of the reduction. For $c, c' \in \mathcal{C}$ and $\Pi_V \subseteq \Pi_{\mathcal{V}}$, let $score(c, c', \Pi_V)$ be the Borda score of c minus the Borda score of c' in $(\mathcal{C}, \Pi_V)$. It is easy to verify that $score(c_x^R, p, \Pi_{\mathcal{V}} \setminus \Pi_{\mathcal{U}}) = 2\kappa x - 1$ and $score(p, c_x^L, \Pi_{\mathcal{V}} \setminus \Pi_{\mathcal{U}}) > 0$ for every $x \in \{1, 2, ..., 3\kappa\}$.

($\Rightarrow$:) Assume that there is an exact 3-set cover $S' \subset S$ of $\mathcal{I}$. Let $\Pi_{S'} = \{\pi_i \mid s_i \in S'\}$. It is easy to verify that $score(c_x^R, p, (\Pi_{\mathcal{V}} \setminus \Pi_{\mathcal{U}}) \cup \Pi_{S'}) = 0$ and $score(p, c_x^L, (\Pi_{\mathcal{V}} \setminus \Pi_{\mathcal{U}}) \cup \Pi_{S'}) > 0$ for every $x \in \{1, 2, ..., 3\kappa\}$. So, after adding all votes in $\Pi_{S'}$, p becomes a winner.

(:$\Leftarrow$) Let Π' be a solution of $\mathcal{E}_{\mathcal{I}}$ and $S' = \{s_i \mid \pi_i \in \Pi\}$. According to the construction, for every $x \in \{1, 2, ..., 3\kappa\}$ each unregistered vote π_i gives $2x$ more points to p than to c_x^R if $c_x \notin s_i$, and gives $2x - 1$ more points to p than to c_x^R if $c_x \in s_i$. Since $score(c_x^R, p, \Pi_{\mathcal{V}} \setminus \Pi_{\mathcal{U}}) = 2\kappa x - 1$, there must be at least $\kappa - 1$ elements $s_i \in S'$ such that $c_x \notin s_i$. As $|S'| \leq \kappa$, S' is an exact 3-set cover of $\mathcal{I}$.

To prove the NP-hardness of the unique-winner model of CCAV, we need only to replace the registered vote $(p, c_1^L, \mathbf{b}, \mathbf{a} \setminus \{c_1^L\})$ with $(p, c_1^L, c_2^L, \mathbf{b}, \mathbf{a} \setminus \{c_1^L, c_2^L\})$.

CCDV. Consider now the nonunique-winner model of CCDV for Borda in single-peaked elections. We create an instance $\mathcal{E}_{\mathcal{I}} = (\mathcal{C}, \Pi_{\mathcal{V}}, p \in \mathcal{C}, R = \kappa)$ as follows.

Candidates $\mathcal{C}$**.** We create in total $9\kappa+1$ candidates. In particular, for each $c_x \in U$, we create two candidates c_x^L and c_x^R. Moreover, we create a set $D = \{d_1, d_2, ..., d_{3\kappa}\}$ of 3κ dummy candidates. Finally, we create a distinguished candidate p.

We shall create votes that are single-peaked with respect to $\rhd$ defined as

$$(c_{3\kappa}^L, c_{3\kappa-1}^L, ..., c_1^L, c_1^R, c_2^R, ..., c_{3\kappa}^R, p, d_1, d_2, ..., d_{3\kappa}).$$

Votes $\Pi_{\mathcal{V}}$**.** We create in total 5κ votes. First, for each $s_i \in S$, we create a vote π_i such that (1) for every $c_x \in U$, if $c_x \in s_i$ then $\pi_i(c_x^L) = 2x - 1$ and $\pi_i(c_x^R) = 2x$; otherwise, $\pi_i(c_x^R) = 2x - 1$ and $\pi_i(c_x^L) = 2x$; (2) $\pi_i(p) = 6\kappa + 1$; and (3) for every $d_y \in D$, where $y \in \{1, 2, ..., 3\kappa\}$, $\pi_i(d_y) = 6\kappa + 1 + y$. In addition, we create a multiset Π_A of $2\kappa - 2$ votes with the same preference. In particular, for each $\pi \in \Pi_A$ it holds that (1) $\pi(p) = 1$; and (2) for every $x \in \{1, 2, ..., 3\kappa\}$, $\pi(d_x) = 2x, \pi(c_x^R) = 6\kappa - 2x + 3$ and $\pi(c_x^L) = 6\kappa+1+x$. Finally, we create a multiset Π_B of 2 votes. In particular, for each $\pi \in \Pi_B$ it holds that (1) $\pi(p) = 1$; and (2) for every $x \in \{1, 2, ..., 3\kappa\}$, $\pi(d_x) = 2x + 1, \pi(c_x^R) = 6\kappa - 2x + 2$ and $\pi(c_x^L) = 6\kappa + 1 + x$.

One can check that all votes created above are single-peaked with respect to $\rhd$. Moreover, $score(c_x^R, p, \Pi_{\mathcal{V}}) = 2\kappa(3\kappa - x + 1) - 1$ and $score(p, c_x^L, \Pi_{\mathcal{V}}) \geq 3$, for every $x \in \{1, 2, ..., 3\kappa\}$ (recall that every element in U occurs in exactly three 3-subsets in S). It remains to prove the correctness of the reduction.

($\Rightarrow$:) Assume that there is an exact 3-set cover $S' \subset S$ of $\mathcal{I}$. Let $\Pi_{S'} = \{\pi_i \mid s_i \in S'\}$. One can check that $score(c_x^R, p, \Pi_{\mathcal{V}} \setminus \Pi_{S'}) = 0$ and $score(p, c_x^L, \Pi_{\mathcal{V}} \setminus \Pi_{S'}) > 0$ for every $x \in \{1, 2, ..., 3\kappa\}$. So p becomes a winner after deleting all votes in $\Pi_{S'}$, implying that $\mathcal{E}_{\mathcal{I}}$ is a YES-instance.

(:$\Leftarrow$) Observe that each vote in $\Pi_A \cup \Pi_B$ ranks p in the top. Therefore, if $\mathcal{E}_{\mathcal{I}}$ is a YES-instance, there is a solution including no vote in $\Pi_A \cup \Pi_B$. Let Π be such a solution. Moreover, let $S' = \{s_i \mid \pi_i \in \Pi\}$. According to the construction, for every $c_x \in U$, $x \in \{1, 2, ..., 3\kappa\}$, a vote $\pi_i \in \Pi$ gives $6\kappa + 1 - 2x$ more points to c_x^R than to p if $c_x \in s_i$; and $6\kappa + 1 - (2x - 1)$ more points to c_x^R than to p if $c_x \notin s_i$. As $score(c_x^R, p, \Pi_{\mathcal{V}}) = 2\kappa(3\kappa - x + 1) - 1$, this implies that for every $c_x \in U$ there are at least $\kappa - 1$ elements $s_i \in S'$ such that $c_x \notin s_i$. From $|S'| \leq \kappa$, it follows that S' is an exact 3-set cover of $\mathcal{I}$.

To prove the unique-winner model, we modify the above reduction slightly. In particular, we create one more vote in Π_A with the same preference and one less vote in Π_B. $\square$

Now we turn our attention to constructive control by adding/deleting votes/candidates for Borda in single-dived elections. In particular, we prove that all these problems are polynomial-time solvable in single-dived elections. The following lemma summarizes a property of Borda scores in single-dived elections which is useful in developing polynomial-time algorithms for the problems studied.

LEMMA 1. *Let $\mathcal{E} = (\mathcal{C}, \Pi_{\mathcal{V}})$ be a single-dived election with respect to a linear order $\rhd$ over $\mathcal{C}$. Then, at least one of $\{\rhd[1], \rhd[m]\}$ is a Borda winner, where $m = |\mathcal{C}|$. Moreover, if for some $1 < i < m$, $\rhd[i]$ is a winner, then the last ranked candidate in each vote is either $\rhd[1]$ or $\rhd[m]$.*

PROOF. For $c \in \mathcal{C}$ and $\pi \in \Pi_{\mathcal{V}}$, let $SC(c, \pi)$ be the score of c obtained from π, and $SC(c) = \sum_{\pi \in \Pi_{\mathcal{V}}} SC(c, \pi)$. Let i be any arbitrary integer such that $1 < i < m$. We partition $\Pi_{\mathcal{V}}$ into three submultisets Π_L, Π_i and Π_R, where Π_L consists of all votes whose last ranked candidates are some $\rhd[i']$ where $i' < i$, Π_R consists of all votes whose last ranked candidates are some $\rhd[i']$ where $i' > i$, and Π_i consists of all the remaining votes. Let $x = |\Pi_L|$ and $y = |\Pi_R|$. To prove the first statement, we need to prove that

$$\max\{SC(\rhd[1]), SC(\rhd[m])\} \geq SC(\rhd[i]).$$

For every vote $\pi \in \Pi_L$, it holds that $SC(\rhd[m], \pi) - SC(\rhd[i], \pi) \geq m - i$ and $SC(\rhd[i], \pi) - SC(\rhd[1], \pi) \leq i - 1$. For every vote $\pi \in \Pi_R$, it holds that $SC(\rhd[1], \pi) - SC(\rhd[i], \pi) \geq i - 1$ and $SC(\rhd[i], \pi) - SC(\rhd[m], \pi) \leq m - i$. For every vote $\pi \in \Pi_i$, it holds that $SC(\rhd[m], \pi) - SC(\rhd[i], \pi) > 0$ and $SC(\rhd[1], \pi) - SC(\rhd[i], \pi) > 0$. Therefore, if $x \geq y$, then $SC(\rhd[m]) - SC(\rhd[i]) \geq \sum_{\pi \in \Pi_L \cup \Pi_i} (SC(\rhd[m], \pi) - SC(\rhd[i], \pi)) - \sum_{\pi \in \Pi_R} (SC(\rhd[i], \pi) - SC(\rhd[m], \pi)) \geq x(m-i) - y(m-i) \geq 0$. If $y \geq x$, we can get that $SC(\rhd[1]) - SC(\rhd[i]) \geq 0$ analogously.

Consider now the second statement. Due to symmetry, assume that the Borda score of $\rhd[m]$ is greater than or equal to that of $\rhd[1]$, and the Borda score of $\rhd[i]$ is equal to that of $\rhd[m]$. That is, both $\rhd[m]$ and $\rhd[i]$ are winners. Due to the proof of the first statement, we know that $x = y$ (otherwise, either $\rhd[m]$ has a greater Borda score than that of $\rhd[i]$ (if $x > y$), or $\rhd[1]$ has a greater Borda score than that of $\rhd[i]$ (if $y > x$)). Moreover, the equality of the Borda scores of $\rhd[i]$ and $\rhd[m]$ implies that every vote in Π_R ranks $\rhd[m]$ in the last position and $\Pi_i = \emptyset$. As a result, $\rhd[1]$ has Borda score at least $x(m-1)$. It follows then that $\rhd[m]$ has Borda score $x(m-1)$ too. Therefore, all votes in Π_L rank $\rhd[1]$ in the last position. □

Due to Lemma 1, the unique winner of a single-dived election must be either the first or the last candidate in $\rhd$. A winner can be every candidate. However, if a candidate between $\rhd[1]$ and $\rhd[m]$ in $\rhd$ is a winner, then each vote either ranks $\rhd[1]$ first and $\rhd[m]$ last, or ranks $\rhd[m]$ first and $\rhd[1]$ last. Due to Lemma 1, we can achieve the following theorem.

In the following, we assume that a harmonious order $\rhd$ is given in the problem instances. This assumption is sound since a harmonious order can be calculated in polynomial-time for single-peaked (or single-dived) elections [3, 12, 16].

THEOREM 2. *CCAV, CCDV, CCAC and CCDC for Borda are polynomial-time solvable in single-dived elections.*

PROOF (NONUNIQUE-WINNER MODEL OF CCAC). Let the given instance be $(\mathcal{C}, \Pi_{\mathcal{V}}, p \in \mathcal{C}, \mathcal{D} \subseteq \mathcal{C} \setminus \{p\}, R)$. In addition let $\rhd$ be a linear order over $\mathcal{C}$ with respect to which $(\mathcal{C}, \Pi_{\mathcal{V}})$ is single-peaked. Let $\rhd'$ be the linear order $\rhd$ restricted to $\mathcal{C} \setminus \mathcal{D}$. If $p \notin \{\rhd'[1], \rhd'[t]\}$, where $t = |\mathcal{C} \setminus \mathcal{D}|$, then due to Lemma 1, we cannot make p a winner by adding candidates (recall that p does not win $(\mathcal{C} \setminus \mathcal{D}, \Pi_{\mathcal{V}}^{\mathcal{C} \setminus \mathcal{D}})$). Assume that p is either the first or the last candidate in $\rhd'$. Due to symmetry, assume that $p = \rhd'[1]$. Again, due to Lemma 1 to make p a winner we cannot add any candidate on the left side of p in $\rhd$. Then, we enumerate all possible candidates $\rhd[i]$ such that either $\rhd[i] = \rhd'[t]$ or $i > j$ where $\rhd[j] = \rhd'[t]$, and for each enumeration we ask if it is possible to add at most $R' = R-1$ (if $\rhd[i] = \rhd'[t]$, replace $R-1$ with R) candidates lying between p and $\rhd[i]$ in $\rhd$ to make p a winner. Due to Lemma 1, we need only to make p have a Borda score no less than that of $\rhd[i]$. For each $c \in \mathcal{C}$ lying between p and $\rhd[i]$ in $\rhd$, let $sg(c) = N(p, c) - N(\rhd[i], c)$, i.e., the increase of the score gap between p and $\rhd[i]$ caused by the addition of c. We order these candidates according to the values of $sg(c)$, from the highest to the lowest. Then, we add the first up to k candidates in the order one-by-one: if after deleting a candidate p becomes a winner, we immediately conclude that the instance is a YES-instance. If after adding the first k candidates in the order p still does not win, the instance is a NO-instance. □

	CCAV	CCDV	CCAC	CCDC
single-peaked	FPT	XP (k) FPT ($k + R$)	XP	XP
single-dived	P	P	P	P

Table 2: Complexity of constructive control by adding/deleting votes/cadidates for Borda$_*$ in k-truncated single-peaked/dived elections, where $* \in \{\uparrow, \downarrow, av\}$. The Parameterized complexity results for CCAV, CCAC and CCDC are with respect to k.

3. K-TRUNCATED SINGLE-PEAKED ELECTIONS

In this section, we study control problems in k-truncated single-peaked/dived elections and investigate how the values of k impact on the complexity of these control problems in this case. A *parameterized problem* is a language $\Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet. The first component is called the *main part* and the second component is called the *parameter*. A parameterized problem is *fixed-parameter tractable* (FPT) if it is solvable in $O(f(k) \cdot |I|^{O(1)})$ time, and is in XP if it is solvable in $O(|I|^{g(k)})$ time, where $|I|$ is the size of the main part, k is the parameter, and $f(k)$ and $g(k)$ are computable functions in k. For further discussion on parameterized complexity, we refer to [13].

We first study k-truncated single-peaked elections. In this setting, we consider first CCAV. It is known that CCAV for Borda in general is NP-hard [31]. Moreover, Liu and Zhu [25] proved that CCAV for Borda in general is W[2]-hard[2] with respect to the solution size R. We have shown in the previous section that CCAV for Borda is NP-hard in single-peaked elections. Now we prove that CCAV for Borda$*$ where $* \in \{\uparrow, \downarrow, av\}$ in k-truncated single-peaked elections is FPT, with respect to k. In particular, we give an integer-linear programming formulation (ILP) with bounded number of variables for the problem. It is known that ILP is FPT with respect to the number of variables [24].

THEOREM 3. *CCAV for Borda$_*$ where $* \in \{\uparrow, \downarrow, av\}$ is FPT in k-truncated single-peaked elections, with respect to k.*

PROOF (UNIQUE-WINNER MODEL). Let the given instance be $\mathcal{I} = (\mathcal{C}, \Pi_{\mathcal{V}}, p \in \mathcal{C}, \Pi_{\mathcal{U}} \subseteq \Pi_{\mathcal{V}}, R)$, and the order with respect to which $(\mathcal{C}, \Pi_{\mathcal{V}})$ is single-peaked be $\rhd$. Let A and B be the sets of the (up to) k candidates lying immediately on the left-side and right-side of p in $\rhd$, respectively, i.e.,

$$A = \{c \in \mathcal{C} \setminus \{p\} \mid 0 < \rhd(p) - \rhd(c) \leq k\}$$

and

$$B = \{c \in \mathcal{C} \setminus \{p\} \mid 0 < \rhd(c) - \rhd(p) \leq k\}.$$

Observe that every YES-instance has an optimal solution including only k-truncated unregistered votes which give p a positive score. Then, due to the definition of k-truncated single-peaked elections, every k-truncated vote in such a solution gives 0 points to all candidates not in $A \cup B \cup \{p\}$. Hence, we need only to focus on

[2]The parameterized complexity class W[2] is a super class of FPT. A problem is W[2]-hard if all problems in W[2] are FPT-reducible to this problem. Unless FPT=W[2], W[2]-hard problems do not admit FPT-algorithms.

the scores of candidates in $A \cup B \cup \{p\}$ given by the unregistered votes. This allows us to formulate the instance into an ILP instance with at most k^{2k+1} variables. Let $\Pi_{\mathcal{U}'}$ be the multiset of all unregistered votes giving p a positive score. We partition $\Pi_{\mathcal{U}'}$ into at most k^{2k+1} multisets, each consisting of all k-truncated unregistered votes in $\Pi_{\mathcal{U}'}$ giving the same score to every candidate in $A \cup B \cup \{p\}$. Let $t = |A \cup B \cup \{p\}|$ and $(c_1 = p, c_2, ..., c_t)$ a fixed order of $A \cup B \cup \{p\}$. We create for each multiset discussed above a variable x_β where β is a t-dimensional integer vector such that the i-th component $\beta[i]$ is the score that every k-truncated vote in this submultiset gives to c_i. Each variable indicates how many k-truncated votes from the corresponding submultiset are included in an optimal solution. Let n_β be the size of the multiset corresponding to x_β. Let $\Pi = \Pi_{\mathcal{V}} \setminus \Pi_{\mathcal{U}}$. For $c \in \mathcal{C}$, let $SC(c, \Pi)$ be the total score of c given by all k-truncated votes in Π. We have the following restrictions:

1. for every variable x_β, we have $0 \leq x_\beta \leq n_\beta$;
2. $\sum_\beta x_\beta \leq R$, where β runs over all possible t-dimensional integer vectors discussed above; and
3. to ensure that p becomes the unique winner, for every $c_i \in A \cup B$, we have that
$$SC(p, \Pi) + \sum_\beta x_\beta \beta[1] > SC(c_i, \Pi) + \sum_\beta x_\beta \beta[i],$$
and for every $c \in \mathcal{C} \setminus (A \cup B \cup \{p\})$, we have
$$SC(p, \Pi) + \sum_\beta x_\beta \beta[1] > SC(c, \Pi),$$
where β runs over all possible t-dimensional integer vectors discussed above.

Due to the result in [24], such an ILP instance can be solved in FPT-time with respect to k. □

Now we study CCDV. In general, it is NP-hard [31]. Moreover, we have shown that it is NP-hard even in single-peaked elections. From the parameterized complexity point of view, whether CCDV for Borda in general is FPT with respect to R remained open. We show now that with respect to the combined parameters k and R (or $k + R$), CCDV for Borda$_*$ is FPT in k-truncated single-peaked elections, where $* \in \{\uparrow, \downarrow, av\}$. As a consequence, for any constant k, CCDV for Borda$_*$ in k-truncated single-peaked elections is FPT with respect to R.

For an election $(\mathcal{C}, \Pi_{\mathcal{V}})$, a positional scoring vector (or correspondence) $\vec{a}$ and a candidate $c \in \mathcal{C}$, let $score(c, (\mathcal{C}, \Pi_{\mathcal{V}}), \vec{a})$ be the score of c in the election calculated according to $\vec{a}$.

THEOREM 4. *CCDV for Borda$_*$ where $* \in \{\uparrow, \downarrow, av\}$ is FPT in k-truncated single-peaked elections with respect to $k + R$.*

PROOF (UNIQUE-WINNER MODEL). Consider first Borda$_\uparrow$. Let $I = (\mathcal{C}, \Pi_{\mathcal{V}}, p \in \mathcal{C}, R)$ be a given instance and $\rhd = (c_1, ..., c_m)$ an order with respect to which $(\mathcal{C}, \Pi_{\mathcal{V}})$ is single-peaked. Moreover, let $m = |\mathcal{C}|$, $n = |\Pi_{\mathcal{V}}|$. Let
$$\vec{a} = \langle m-1, m-2, ..., m-k, 0, ..., 0\rangle$$
and $\ell = (m-1) \cdot n$.

The framework of the algorithm is to first break down the given instance into polynomially many subinstances, and then solve each subinstance by formulating it into an ILP instance whose number of variables is bounded by a function of k and R. In particular, the algorithm breaks down I into ℓ subinstances, each of which takes I and an integer s such that $0 < s \leq \ell$ as the input, and asks whether there are at most R many k-truncated votes in $\Pi_{\mathcal{V}}$ whose deletion results in p having a score at least s and every other candidate having a score less than s. Clearly, I is a YES-instance if and only if at least one of the subinstances is a YES-instance. Moreover, if each subinstance is solvable in FPT-time, so is I. Let $I' = (I, s)$ be a subinstance. In the following, we show how to solve I' in FPT-time.

Let $A = \{c \in \mathcal{C} \setminus \{p\} \mid score(c, (\mathcal{C}, \Pi_{\mathcal{V}}), \vec{a}) \geq s\}$. Observe that if I' is a YES-instance, then there is an optimal solution which consists of only k-truncated votes that rank at least one candidate in A in the top-k positions. Hence, for every YES-instance I' it holds that $|A| \leq k \cdot R$. As deleting votes does not increase the scores of candidates, all candidates in $\mathcal{C} \setminus (A \cup \{p\})$ have score less than s in the final election no matter which k-truncated votes are deleted. As a consequence, we need only to consider the impact of deletions of k-truncated votes on the scores of candidates in $A \cup \{p\}$. We partition $\Pi_{\mathcal{V}}$ into multisets, each consisting of all k-truncated votes giving exactly the same scores to all candidates in $A \cup \{p\}$. That is, two k-truncated votes π and π' are in the same multiset if and only if for every $c \in A \cup \{p\}$ such that $\pi(c) \leq k$, it holds that $\pi(c) = \pi'(c)$. Clearly, the number of submultisets is bounded by a function of k and R.

Now we are ready to give the ILP formulation for I'. In particular, we create a variable for each submultiset discussed above, indicating how many k-truncated votes from this submultiset are deleted. Precisely, for each submultiset we create a variable $x(B, f)$ where $B \subseteq A \cup \{p\}$ and $f : B \mapsto \{1, 2, ..., k\}$ is a mapping such that every k-truncated vote in the submultiset ranks each candidate $c \in B$ in the $f(c)$-th position and ranks every candidate in $(A \cup \{p\}) \setminus B$ in some k'-th position with $k' > k$. Therefore, every k-truncated vote in the submultiset corresponding to a variable $x(B, f)$ gives $\vec{a}[f(c)]$ points to every candidate $c \in B$ and 0 points to every candidate $c \in (A \cup \{p\}) \setminus B$. Let $n(B, f)$ be the number of k-truncated votes in the submultiset corresponding to the variable $x(B, f)$. The restrictions are as follows.

1. Since we are allowed to delete in total at most R many k-truncated votes, we have
$$\sum x(B, f) \leq R,$$
where $x(B, f)$ runs over all variables defined above.
2. For every variable $x(B, f)$, we have
$$0 \leq x(B, f) \leq n(B, f).$$
3. To ensure the final score of p is at least s, we have
$$score(p, (\mathcal{C}, \Pi_{\mathcal{V}}), \vec{a}) - \sum x(B, f) \cdot \vec{a}[(f(p)] \geq s,$$
where $x(B, f)$ runs over all variables corresponding to the submultsets of k-truncated votes which rank p in the top-k positions.
4. To ensure the final score of every candidate in A is less than s, for every $c \in A$ we have
$$score(c, (\mathcal{C}, \Pi_{\mathcal{V}}), \vec{a}) - \sum x(B, f) \cdot \vec{a}[f(c)] < s,$$
where $x(B, f)$ runs over all variables corresponding to the submultsets of k-truncated votes which rank c in the top-k positions.

For Borda$_\downarrow$ (Borda$_{av}$), we need only to replace $\vec{a}$ with $\langle k, k-1, ..., 1, 0, ..., 0\rangle$ ($\langle m+k-1, m+k-3, ..., m-k+1, 0, ..., 0\rangle$), and redefine $\ell = k \cdot n$ ($\ell = (m+k-1) \cdot n$). □

We are unable to show the fixed-parameter tractability of CCDV as we did for CCAV with respect to the single parameter k. The reason is that in this case an optimal solution may contain votes that give 0 points to p, and hence, we cannot limit our attention to a set of candidates whose cardinality is bounded by a function of k. Nevertheless, we can show that with respect to k the problem is in XP, by a dynamic programming algorithm.

THEOREM 5. *CCDV for Borda$_*$ where $* \in \{\uparrow, \downarrow, av\}$ in k-truncated single-peaked elections is in XP with respect to k.*

PROOF. Let $I = (\mathcal{C}, \Pi_\mathcal{V}, p \in \mathcal{C}, \rhd, R), m, n, \vec{a}, \rhd$ and ℓ be as defined in the proof of Theorem 4. We develop an XP-algorithm for the problem stated in the theorem as follows. Without loss of generality, assume that $p = c_z$ for some $k \leq z \leq m-k$, i.e., there are at least $k-1$ candidates lying on both sides of p in $\rhd$. Indeed, if this was not the case, we could add $2k-2$ dummy candidates ranked after all other candidates in all k-truncated votes, and put $k-1$ of them one-by-one on the leftmost positions of $\rhd$ and put the remaining of them one-by-one on the rightmost positions of $\rhd$. For each $\pi_v \in \Pi_\mathcal{V}$, let $+(\pi_v)$ be the set of candidates whose scores decrease after the deletion of π_v, i.e., $+(\pi_v) = \{c \in \mathcal{C} \mid \pi_v(c) \leq k\}$. An observation is that the candidates in $+(\pi_v)$ lie consecutively in $\rhd$. For ease of exposition, let $c(\pi_v, 1), c(\pi_v, 2), ..., c(\pi_v, k)$ denote the candidates in $+(\pi_v)$ such that $\rhd(c(\pi_v, x)) < \rhd(c(\pi_v, y))$ for every $1 \leq x < y \leq k$.

Let Π_1 be the multiset of all k-truncated votes π_v such that $p \in +(\pi_v)$. Deleting a k-truncated vote in Π_1 does not affect the score of any candidate c_i such that $|i - z| \geq k$. Moreover, deleting any k-truncated vote not in Π_1 does not change the score of p. The algorithm breaks down I into $n \cdot (\ell+1)^{2k-1}$ subinstances, each of which takes I together with a $(2k-1)$-dimensional integer vector $\langle b_{z-k+1}, b_{z-k+2}, ..., b_z, ..., b_{z+k-1}\rangle$ and a nonnegative integer $t \leq R$ as the input, where each $0 \leq b_x \leq \ell$ for $z-k+1 \leq x \leq z+k-1$, and asks if there is a submultiset $\Pi_\mathcal{T} \subseteq \Pi_\mathcal{V}$ such that

1. $|\Pi_\mathcal{T}| \leq R$;
2. $|\Pi_\mathcal{T} \cap \Pi_1| = t$;
3. for every $c_x, z-k+1 \leq x \leq z+k-1$,
$$score(c_x, (\mathcal{C}, \Pi_\mathcal{T} \cap \Pi_1), \vec{a}) = b_x;$$
4. for every $c_x \neq p$,
$$score(c_x, (\mathcal{C}, \Pi_\mathcal{V} \setminus \Pi_\mathcal{T}), \vec{a}) < score(c_z, (\mathcal{C}, \Pi_\mathcal{V}), \vec{a}) - b_z.$$

The algorithm first focuses on $\Pi_\mathcal{T} \cap \Pi_1$, i.e., the part of the solution in Π_1. Determining whether there is a $\Pi' \subseteq \Pi_1$ such that $\Pi' = \Pi_\mathcal{T} \cap \Pi_1$ and Π' satisfies Conditions (2) and (3) can be done in XP-time (in fact FPT-time) using ILP-based algorithms. Precisely, we say two k-truncated votes π_1, π_2 in Π_1 have the same type if their rankings of the first k candidates are identical, i.e., $\pi_1[i] = \pi_2[i]$ for every $1 \leq i \leq k$. There are in total at most $k^2 2^k$ different types. For each type, we assign a variable indicating how many k-truncated votes of this type are in Π'. The restrictions are analogous to the ones in the proof of Theorem 3.

Now we show how to calculate $\Pi_\mathcal{T} \setminus \Pi_1$. Let
$$sc_p = score(b_z, (\mathcal{C}, \Pi_\mathcal{V}), \vec{a}) - b_z.$$
In addition we reset $R := R - t$. For each $i \in \{1, ..., z-k, z+k, ..., m\}$, define $b_i = 0$. Let $(\pi_1, \pi_2, ..., \pi_u)$ be an order of all k-truncated votes in $\Pi_\mathcal{V} \setminus \Pi_1$ such that $c(\pi_x, 1)$ is not on the right side of $c(\pi_y, 1)$ in $\rhd$ for every $1 \leq x < y \leq u$. Our algorithm is based on dynamic programming. In particular, we maintain a table $DT(i, R', s_1, s_2, ..., s_k)$ where i, R' are integers such that $1 \leq i \leq u$, $0 \leq R' \leq \min\{R, i\}$, and each $s_x, 1 \leq x \leq k$ is an integer between 0 to ℓ. We define $DT(i, R', s_1, s_2, ..., s_k) = 1$ if and only if there exists $\Pi_\mathcal{H} \subseteq \{\pi_1, ..., \pi_i\}$ such that

1. $|\Pi_\mathcal{H}| = R'$;
2. for each $1 \leq x \leq k$, $score(c(\pi_i, x), (\mathcal{C}, \Pi_\mathcal{H}), \vec{a}) = s_x$; and
3. for every candidate c_x lying on the left side of $c(\pi_i, 1)$ in $\rhd$, $score(c_x, (\mathcal{C}, \Pi_\mathcal{H}), \vec{a}) > score(c_x, (\mathcal{C}, \Pi_\mathcal{V}), \vec{a}) - b_x - sc_p$.

The initialization of the dynamic table is as follows:

1. $DT(1, 0, s_1, ..., s_k) = 1$ if and only if $s_x = 0$ for every $1 \leq x \leq k$, and for every candidate $c_x \in \mathcal{C} \setminus \{p\}$ it holds that $score(c_x, (\mathcal{C}, \Pi_\mathcal{V}), \vec{a}) - b_x - sc_p < 0$;
2. $DT(1, 1, s_1, ..., s_k) = 1$ if and only if for every $1 \leq x \leq k$ it holds that $s_x = score(c(\pi_1, x), (\mathcal{C}, \{\pi_1\}), \vec{a})$, and for every candidate c_x lying on the left side of $c(\pi_1, 1)$ it holds that
$$score(c_x, (\mathcal{C}, \{\pi_1\}), \vec{a}) > score(c_x, (\mathcal{C}, \Pi_\mathcal{V}), \vec{a}) - b_x - sc_p.$$

We use the following relation to update the table:
$DT(i, R', s_1, ..., s_k) = 1$ if and only if

- $DT(i-1, R', s_1, ..., s_k) = 1$ (only when $R' \leq i-1$); or
- $DT(i-1, R'-1, s'_1, ..., s'_k) = 1$ where for every $1 \leq x \leq k$, $s'_x = s_x - score(c(\pi_i, x), (\mathcal{C}, \{\pi_i\}), \vec{a})$.

Clearly, the subinstance is a YES-instance if there exists $DT(i = u, R' \leq R, s_1, ..., s_k) = 1$. As we have in total at most $n^2 \cdot (\ell+1)^k$ entries to calculate, the running time of the above dynamic programming algorithm is bounded by $O(n^2 \cdot (\ell+1)^k)$. As we have at most $n \cdot (\ell+1)^{2k-1}$ subinstances, the whole running time of the algorithm is $O(n^3 \cdot (\ell+1)^{3k-1})$, an XP-algorithm. □

Now we study control by adding/deleting candidates in k-truncated single-peaked elections. Faliszewski et al. [19] studied CCAC and CCDC for Plurality in single-peaked elections, and proved that they are polynomial-time solvable. Recall that Plurality can be regarded exactly as Borda$\uparrow$, Borda$_\downarrow$ and Borda$_{av}$ in 1-truncated elections. We extend their results by showing that CCAC and CCDC for Borda$_\uparrow$, Borda$_\downarrow$ and Borda$_{av}$ in k-truncated single-peaked elections are all polynomial-time solvable for any constant k.

Before presenting our algorithms, let's first study a property. In general, it states that the score of every candidate c is determined by the $k-1$ candidates lying on the left-side of c in $\rhd$, the $k-1$ candidates lying on the right-side of c in $\rhd$, and c herself.

LEMMA 2. *Let $(\mathcal{C}, \Pi_\mathcal{V})$ be a k-truncated single-peaked election with respect to a linear order $\rhd$ over $\mathcal{C}$. Then, for every Borda$_*$ where $* \in \{\uparrow, \downarrow, av\}$, it holds that*
$$score(c, (\mathcal{C}, \Pi_\mathcal{V}), Borda_*) = score(c, (C, \Pi_\mathcal{V}^C), Borda_*),$$
where $C = \{c' \in \mathcal{C} \mid \rhd(c) - k + 1 \leq \rhd(c') \leq \rhd(c) + k - 1\}$.

Now we are ready to give our algorithms for CCAC and CCDC for Borda$_*$ for every $* \in \{\uparrow, \downarrow, av\}$ in k-truncated single-peaked elections.

THEOREM 6. *CCAC and CCDC for Borda$_*$ for every $* \in \{\uparrow, \downarrow, av\}$ in k-truncated single-peaked elections are XP with respect to k.*

PROOF (UNIQUE-WINNER MODEL FOR CCDC). We first describe an algorithm for CCDC in k-truncated single-peaked elections for the following voting correspondence characterized by k rational numbers $\alpha(1),...,\alpha(k)$: each vote gives $\alpha(i)$ points to the candidate in the i-th position where $1 \leq i \leq k$, and gives 0 points to every candidate in the i'-th position where $i' > k$, and the winners are the ones with the highest total score.

Let $I = (\mathcal{C}, \Pi_\mathcal{V}, p \in \mathcal{C}, R)$ be a given instance and $\rhd$ an order with respect to which all votes in $\Pi_\mathcal{V}$ are single-peaked. Let $m = |\mathcal{C}|$. If $R \leq 2k$ or $m \leq R + k$, the instance can be solved in $O(m^{O(k)})$-time by enumerating all potential solutions. Assume now that $R > 2k$ and $m > R + k$. Let $X = \{c \in \mathcal{C} \mid \rhd(c) < \rhd(p)\}$ and $Y = \{c \in \mathcal{C} \mid \rhd(c) > \rhd(p)\}$ be the sets of candidates lying on the left side of p and right side of p in $\rhd$, respectively. The algorithm first breaks down I into polynomially many subinstances, each taking the original instance, two sets $A \subseteq X, B \subseteq Y$ such that $|A| = \min\{k, |X|\}$, $|B| = \min\{k, |Y|\}$ and an integer $0 \leq R' \leq R$ as the input. Let a (resp. b) be the leftmost (resp. rightmost) candidate of A (resp. B) in $\rhd$, i.e., a (resp. b) is the candidate in A (resp. B) such that $\rhd(a) \leq \rhd(c)$ for every $c \in A$ (resp. $\rhd(b) \geq \rhd(c)$ for every $c \in B$). Moreover, let

$$D = \{c \in \mathcal{C} \setminus (A \cup B \cup \{p\}) \mid \rhd(a) < \rhd(c) < \rhd(b)\}.$$

The subinstance asks whether there exists $C \subseteq \mathcal{C} \setminus \{p\}$ such that

1. $|C| = R'$;
2. $(A \cup B) \cap C = \emptyset$;
3. $D \subseteq C$;
4. p wins the election after deleting all candidates in C.

In general, the subproblem aims to find a potential solution of size exactly R', and A (resp. B) includes exactly the k candidates lying consecutively on the left side (right side) of p in $\rhd$ in the final election. Clearly, the original instance is a YES-instance if and only if one of the subinstances is a YES-instance. Moreover, as we have at most m^{2k+1} subinstances, the original instance is solvable in XP-time if each subinstance is solvable in XP-time. Let $I' = (I, A, B, R')$ be a subinstance. Clearly, if $|D| > R'$, I' is a NO-instance. Otherwise, we consider the election after deleting all candidates in D. Then, due to Lemma 2, the final score of p is known. Let s be the final score of p. As deleting a candidate never decreases the score of every other candidate, all candidates that have score more than s have to be deleted in order to make p win the election. Hence, if there is such a candidate in $A \cup B$, I' is a NO-instance. Otherwise, we delete those candidates which have scores more than s iteratively (notice that it may happen that after deleting a candidate, a candidate having score less than s previously has score more than s). If more than R' candidates are deleted, I' is a NO-instance; otherwise I' is a YES-instance.

Borda$_\downarrow$ falls into the category of such voting correspondences. In particular, by setting $\alpha(i) = k - (i - 1)$ the above algorithm solves CCDC for Borda$_\downarrow$ in k-truncated single-peaked elections.

Consider now Borda$_\uparrow$. At first glance, the above algorithm does not apply to Borda$_\uparrow$, since Borda$_\uparrow$ calculates winners with respect to $f(1), f(2), ..., f(k)$ where $f(i) = m - i$ and m is the number of candidates. Hence, candidate deletion changes the values of each $f(i)$. Nevertheless, recall that the subproblem defined above seeks a solution C of size exactly R'. Therefore, if we have m candidates in the original instance, we have exactly $m - R'$ candidates in the final election. Due to this, by setting $\alpha(i) = m - R' - i$ for every $1 \leq i \leq k$ the above algorithm solves CCDC for Borda$_\uparrow$ in k-truncated single-peaked elections. Analogously, by setting $\alpha(i) = m - R' + k - 2i + 1$ the above algorithm solves CCDC for Borda$_{av}$ in k-truncated single-peaked elections. □

Now we study control problems for Borda$_*$ in k-truncated single-dived elections. An observation is that Lemma 1 still holds in this case.

THEOREM 7. *CCAV, CCDV, CCAC and CCDC for Borda$_*$ in k-truncated single-dived elections are polynomial-time solvable, where $* \in \{\uparrow, \downarrow, av\}$.*

4. RELATED WORK

Our NP-hardness results of CCAV and CCDV for Borda in single-peaked elections are most related to [8, 19], where many voting problems which are NP-hard in general are shown to be polynomial-time solvable in single-peaked elections. For weighted elections, Faliszewski et al. [19] obtained a dichotomy results for constructive coalition weighted manipulation with three candidates for positional scoring correspondences. Their results imply that constructive coalition weighted manipulation with three candidates for Borda is polynomial-time solvable. Brandt et al. [8] proved that constructive coalition weighted manipulation with four candidates for Borda is NP-hard. Our NP-hardness results clearly extend to weighted elections.

Our work concerning k-truncated votes is related to the work by Menon and Larson[27], where they studied the complexity of weighted manipulation and bribery problems in k-truncated single-peaked elections. In particular, they proved that there are bribery and manipulation problems which are polynomial-time solvable in single-peaked elections, but become NP-hard when k-truncated votes exist. In this paper, we mainly study control problems. Moreover, we are solely concerned with unweighted elections. Finally, our primary goal is to investigate how the parameter k impacts the parameterized complexity of control problems. However, Menon and Larson[27] are mainly concerned with how the complexity of weighted manipulation and bribery problems changes in single-peaked elections, k-truncated elections and general elections.

Fitzsimmons and Hemaspaandra [21] explored the complexity of many voting problems in single-peaked elections with partial votes, i.e., votes with ties. In particular, they studied constructive coalition manipulation problem for numerous voting correspondences. It should be pointed out that Fitzsimmons and Hemaspaandra [21] studied four different models of single-peaked elections with partial votes. We refer to [21] for further details. In CCAV and CCDV, we do not modify votes. Hence, from purely complexity point of view, in these two problems k-truncated votes can be regarded as partial votes, i.e., the unranked candidates in a vote are less preferred to ranked candidates and the vote is indifferent between the unranked candidates. Nevertheless, in all models studied in [21], ties may occur in several places, but not necessarily only in the last $m - k$ positions, where m is the number of candidates. Moreover, in k-truncated single-peaked elections, every vote ranks exactly k candidates. This is not the case in each of the four models studied in [21]. In addition, in CCAC and CCDC, the associated complete vote to each k-truncated vote is essential in the definition of the problems studied in this paper. Finally, we consider mainly control problems, while Fitzsimmons and Hemaspaandra [21] studied manipulation problems.

Acknowledgments

The author would like to thank the anonymous AAMAS referees for their helpful comments.

REFERENCES

[1] K. J. Arrow. A difficulty in the concept of social welfare. *J. Polit. Econ.*, 58(4):328–346, 1950.

[2] J. J. Bartholdi III, C. A. Tovey, and M. A. Trick. How hard is it to control an election? *Math. Comput. Model.*, 16(8-9):27–40, 1992.

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

[4] D. Baumeister, P. Faliszewski, J. Lang, and J. Rothe. Campaigns for lazy voters: truncated ballots. In *AAMAS*, pages 577–584, 2012.

[5] N. Betzler, R. Niedermeier, and G. J. Woeginger. Unweighted coalitional manipulation under the Borda rule is NP-hard. In *IJCAI*, pages 55–60, 2011.

[6] N. Betzler, A. Slinko, and J. Uhlmann. On the computation of fully propotional representation. *J. Artif. Intell. Res.*, 47:475–519, 2013.

[7] D. Black. On the rationale of group decision-making. *J. Polit. Econ.*, 56(1):23–34, 1948.

[8] F. Brandt, M. Brill, E. Hemaspaandra, and L. A. Hemaspaandra. Bypassing combinatorial protections: Polynomial-time algorithms for single-peaked electorates. *J. Artif. Intell. Res.*, 53:439–496, 2015.

[9] J. Chen, P. Faliszewski, R. Niedermeier, and N. Talmon. Elections with few voters: Candidate control can be easy. In *AAAI*, pages 2045–2051, 2015.

[10] J. Davies, G. Katsirelos, N. Narodytska, and T. Walsh. An empirical study of Borda manipulation. In *COMSOC*, pages 91–102, 2010. http://arxiv.org/abs/1007.5104.

[11] J. Davies, G. Katsirelos, N. Narodytska, T. Walsh, and L. Xia. Complexity of and algorithms for the manipulation of Borda, Nanson's and Baldwin's voting rules. *Artif. Intell.*, 217:20–42, 2014.

[12] J. P. Doignon and J. C. Falmagne. A polynomial time algorithm for unidimensional unfolding representations. *J. Algorithms.*, 16(2):218–233, 1994.

[13] R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

[14] P. Emerson. The original Borda count and partial voting. *Soc. Choice. Welf.*, 40(2):353–358, 2013.

[15] G. Erdélyi, M. Lackner, and A. Pfandler. Computational aspects of nearly single-peaked electorates. In *AAAI*, pages 283–289, 2013.

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

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

[18] P. Faliszewski, E. Hemaspaandra, and L. A. Hemaspaandra. Weighted electoral control. *J. Artif. Intell. Res.*, 52:507–542, 2015.

[19] P. Faliszewski, E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. The shield that never was: Societies with single-peaked preferences are more open to manipulation and control. *Inf. Comput.*, 209(2):89–107, 2011.

[20] P. Faliszewski and J. Rothe. Control and bribery in voting. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 7, pages 146–168. Cambridge University Press, 2016.

[21] Z. Fitzsimmons and E. Hemaspaandra. Modeling single-peakedness for votes with ties. In *STAIRS*, pages 63–74, 2016.

[22] T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoret. Comput. Sci.*, 38:293–306, 1985.

[23] E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. Anyone but him: The complexity of precluding an alternative. *Artif. Intell.*, 171(5-6):255–285, 2007.

[24] H. W. Lenstra. Integer programming with a fixed number of variables. *Math. Oper. Res.*, 8(4):538–548, 1983.

[25] H. Liu and D. Zhu. Parameterized complexity of control by voter selection in Maximin, Copeland, Borda, Bucklin, and Approval election systems. *Theor. Comput. Sci.*, 498:115–123, 2013.

[26] A. Loreggia, N. Narodytska, F. Rossi, K. B. Venable, and T. Walsh. Controlling elections by replacing candidates or votes. In *AAMAS*, pages 1737–1738, 2015.

[27] V. Menon and K. Larson. Reinstating combinatorial protections for manipulation and bribery in single-peaked and nearly single-peaked electorates. In *AAAI*, pages 565–571, 2016.

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

[29] S. Nitzan and A. Rubinstein. A further characterization of Borda ranking method. *Public. Choice.*, 36(1):183–158, 1981.

[30] B. Reilly. Social choice in the south seas: Electoral innovation and the Borda count in the pacific island countries. *Int. Polit. Sci. Rev.*, 23(4):355–372, 2003.

[31] N. F. Russell. Complexity of control of Borda count elections. Master's thesis, Rochester Institute of Technology, 2007.

[32] Y. Yang. Manipulation with bounded single-peaked width: A parameterized study. In *AAMAS*, pages 77–85, 2015.

[33] Y. Yang and J. Guo. The control complexity of r-Approval: from the single-peaked case to the general case. In *AAMAS*, pages 621–628, 2014.

[34] Y. Yang and J. Guo. Controlling elections with bounded single-peaked width. In *AAMAS*, pages 629–636, 2014.

[35] Y. Yang and J. Guo. How hard is control in multi-peaked elections: A parameterized study. In *AAMAS*, pages 1729–1730, 2015.

[36] Y. Yang and J. Guo. Exact algorithms for weighted and unweighted Borda manipulation problems. *Theor. Comput. Sci.*, 622:79–89, 2016.