The Control Complexity of r-Approval: from the Single-Peaked Case to the General Case

Yongjie Yang
Universität des Saarlandes
Campus E 1.7, D-66123 Saarbrücken, Germany
yyongjie@mmci.uni-saarland.de

Jiong Guo
Universität des Saarlandes
Campus E 1.7, D-66123 Saarbrücken, Germany
jguo@mmci.uni-saarland.de

ABSTRACT

A natural generalization of the single-peaked elections is the k-peaked elections, where at most k peaks are allowed in each vote. Motivated by $\mathcal{NP}$-hardness in general and polynomial-time solvability in single-peaked elections, we aim at establishing a complexity dichotomy of several control problems for r-approval voting in k-peaked elections with respect to k. It turns out that most $\mathcal{NP}$-completeness results in general also hold in k-peaked elections, even for $k = 2, 3$. On the other hand, we derive polynomial-time algorithms for certain control problems for $k = 2$. In addition, we also study the problems from the viewpoint of parameterized complexity and achieve both $\mathcal{FPT}$ and $\mathcal{W}$-hardness results. Several of our results apply to approval voting and sincere-strategy preference-based approval voting as well.

Categories and Subject Descriptors

F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity; G.2.1 [**Combinatorics**]: Combinatorial algorithms; J.4 [**Computer Applications**]: Social Choice and Behavioral Sciences

Keywords

single-peaked; approval voting; control; parameterized complexity; kernel

1. INTRODUCTION

Voting is a common method for preference aggregation and collective decision-making, and has applications in political elections, multi-agent systems, web spam reduction, etc. However, by the Arrow's impossibility theorem [1], there is no voting system which satisfies a certain set of desirable criteria (see [1] for the details) when more than two candidates are involved. One possible way to bypass the Arrow's impossibility theorem is to restrict the domain of the preferences, for instance, the single-peaked domain introduced by Black [4]. Intuitively, in a single-peaked election, one can order the candidates from left to right such that every voter's preferences increase first and then decrease after some point as the candidates are considered from left to right.

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

Recently, the complexity of various voting problems in single-peaked elections has been attracting attention of many researchers from both theoretical computer science and social choice communities [6, 13, 15, 16, 23]. It turned out that many voting problems being $\mathcal{NP}$-hard in general become polynomial-time solvable when restricted to single-peaked elections. However, most elections in practice are not purely single-peaked, which motivates researchers to study more general models of elections. We refer readers to [7, 8, 9, 11, 14] for some variations of single-peaked model. In this paper, we consider a natural generalization of single-peaked elections, where more than one peak may occur in each vote. This generalization might be relevant for many real-world applications. For example, consider a group of people who are willing to select a special day for an event. In this setting, each voter may have several special days which he/she prefers for some reason, and the longer the other days away from these favorite days, the less they are preferred by the voter. We call this generalization k-peaked elections.

In this paper, we mainly study control problems for r-approval voting restricted to k-peaked elections. In a control attack, there is an external agent (e.g., the chairman in an election) who is willing to influence the results of the election by doing some tricks. There could be two goals that the external agent wants to reach. One goal is to make some distinguished candidate win the election. The other goal could be to make someone lose the election. The former case is called a *constructive control* and the latter case is called a *destructive control*. Moreover, the tricks involved in a control attack include adding some new, unregistered votes to the registered votes, deleting votes from the registered votes, adding new candidates to the election or deleting candidates from the election. We refer readers to [12, 18, 19] for further information on control attacks.

Approval voting is one of the most famous voting systems and has been extensively studied both in theory and in practice. In an *approval voting*, we are given a set $\mathcal{C}$ of *candidates* and a set $\mathcal{V}$ of *voters*. Each voter approves or disapproves every candidate $c \in \mathcal{C}$. The system selects a candidate who is approved by the most voters as a winner. An *r-approval voting* is a variant of approval voting, where each voter v casts a vote π_v, defined as a linear order (a transitive, antisymmetric, and total binary relationship) over the candidates, and approves the candidates ordered in the top r positions in π_v and disapproves all other candidates. Particularly, 1-approval is also called *plurality*. Another prominent variant related to approval voting is the *sincere-strategy preference-based approval voting* (SP-AV for short), proposed by Brams and Sanver [5]. In an SP-AV election, each voter provides both a linear order of the candidates and a subset C of candidates such that the candidates are approved according to C, and the "admissible" and "sincere" properties should be fulfilled (see [5, 12] for the precise definition).

In the following, we consider only constructive control. Hemaspaandra et al. [18] proved that the control problems by adding/deleting votes for approval voting are $\mathcal{NP}$-complete. The proofs can be adapted to show the $\mathcal{NP}$-completeness of control by adding/deleting votes in SP-AV [12]. Lin [21] proved that control by adding votes in 4-approval and control by deleting votes in 3-approval are both $\mathcal{NP}$-complete, while control by adding votes in 3-approval and control by deleting votes in 2-approval are polynomial-time solvable. As for the control by modification of candidates, approval voting turned out to be immune[1] to control by adding candidates and polynomial-time solvable for control by deleting candidates [18]. However, the control problems by adding/deleting candidates are $\mathcal{NP}$-complete for r-approval, even when degenerated to 1-approval [19]. The $\mathcal{NP}$-completeness also holds for control by adding/deleting candidates in SP-AV [12]. Recently, control in approval voting and r-approval voting have also been considered with respect to single-peaked elections. Faliszewski et al. [15] proved that the control problems by adding/deleting votes in approval are polynomial-time solvable in single-peaked elections[2]. Moreover, the control problems by adding/deleting candidates for 1-approval are polynomial-time solvable in single-peaked elections [15].

Motivated by the $\mathcal{NP}$-completeness in the general case and the polynomial-time solvability in the single-peaked case, we study the complexity of control problems for r-approval voting in k-peaked elections with respect to various values of k, aiming at exploring the complexity border for these control problems. Faliszewski et al. [14] studied a nearly single-peaked model which is called Swoon-SP and can be considered as a special case of 2-peaked elections. They proved that the control problems by adding/deleting candidates for 1-approval are $\mathcal{NP}$-complete when restricted to Swoon-SP elections, implying the $\mathcal{NP}$-completeness of these problems in 2-peaked elections. We complement their results by studying the adding/deleting votes case. Our findings are summarized in Table 1. In particular, we show that, control by adding votes in r-approval with r being a constant is polynomial-time solvable in 2-peaked elections, but $\mathcal{NP}$-complete in k-peaked elections for $k \geq 3$. Meanwhile, if r is not a constant, then control by adding votes in r-approval in 2-peaked elections becomes $\mathcal{NP}$-complete. Moreover, the deleting votes case turns out to be $\mathcal{NP}$-complete for k-peaked elections with $k \geq 2$, even for r being a constant.

In addition, we present some results for r-approval control problems with respect to their parameterized complexity. Recently, many voting problems have been studied from the perspective of parameterized complexity. See [2] for an overview. A *parameterized problem* is a language $L \subseteq \Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet. The first component $I \in \Sigma$ is called the *main part* of the problem while the second component $\kappa \in \mathbb{N}$ is called the *parameter*. Downey and Fellows [10] established the parameterized complexity theory, where the class $\mathcal{FPT}$ (stands for fixed-parameter tractable) includes all parameterized problems which admit $O(f(\kappa) \cdot |I|^{O(1)})$-time algorithms. Here $f(\kappa)$ is a computable function. Another important parameterized complexity class is $\mathcal{W}[1]$ which is the basic class for showing fixed-parameter intractability results. A problem is $\mathcal{W}[1]$-hard if all problems in $\mathcal{W}[1]$ are $\mathcal{FPT}$-reducible to the problem. We can show a problem being $\mathcal{W}[1]$-hard by giving an $\mathcal{FPT}$-reduction from another $\mathcal{W}[1]$-hard problem.

Given two parameterized problems Q and Q', an $\mathcal{FPT}$-*reduction* from Q to Q' is an algorithm that takes as input an instance (I, κ) of Q and outputs an instance (I', κ') of Q' such that

(1) the algorithm runs in $f(\kappa) \cdot |I|^{O(1)}$ time, where f is a computable function in κ;

(2) $(I, \kappa) \in Q$ if and only if $(I', \kappa') \in Q'$; and

(3) $\kappa' \leq g(\kappa)$, where g is a computable function in κ.

Liu et al. [22] proved that control by adding votes in approval voting is $\mathcal{W}[1]$-hard and control by deleting votes in approval voting is $\mathcal{W}[2]$-hard[3], with the numbers of added and deleted votes as parameters, respectively. In addition, they proved that control by adding candidates in 1-approval is $\mathcal{W}[2]$-hard, with the number of added candidates as the parameter. Betzler and Uhlmann [3] complemented the results in [22] by proving that control by deleting candidates in 1-approval is $\mathcal{W}[2]$-hard, with the number of deleted candidates as the parameter.

We extend the above results to k-peaked elections by showing that control by deleting candidates in 1-approval restricted to 3-peaked elections is $\mathcal{W}[1]$-hard with the number of deleted candidates as the parameter. For the general elections, we present two $\mathcal{FPT}$ results for control by adding/deleting votes in r-approval voting, with r being a constant. Here, the parameter is the number of votes added/deleted.

Remarks: All our results apply to both unique-winner and nonunique-winner models. For the sake of clarity, our proofs and algorithms are solely based on the unique-winner model. The corresponding results for the nonunique-winner model can be derived in the similar way. Our $\mathcal{NP}$-completeness results hold for approval voting and SP-AV as well. For $\mathcal{NP}$-completeness results, we present only the $\mathcal{NP}$-hardness proofs, since all problems studied are clearly in $\mathcal{NP}$.

2. PRELIMINARIES

Multisets. A *multiset* $S = \{s_1, s_2, ..., s_{|S|}\}$ is a generalization of a set where objects are allowed to appear more than once, that is, $s_i = s_j$ is allowed for $i \neq j$. An *element* of S is one copy of some object. We use $s \in_+ S$ to denote that s is an element of S. The size of S, denoted by $|S|$, is the number of elements in S. For two multisets A and B, we use $A \uplus B$ to denote the multiset containing all elements from A and B, and use $A \ominus B$ to denote the multiset containing, for each object s, $\max\{0, n_1 - n_2\}$ copies of s, where n_1 and n_2 denote the numbers of copies of s in A and B, respectively. For example, for $A = \{1, 1, 1, 2, 3, 3, 4\}$ and $B = \{1, 2, 3\}$,

[1]A voting system is immune to a control behavior if one cannot make a candidate who is not a winner become a final winner by imposing the strategic behavior on the election.

[2]In [15], for the approval voting, an election is single-peaked if there is an order of the candidates such that each voter's approved candidates are contiguous within the order.

[3]A problem is $\mathcal{W}[2]$-hard if all problems in $\mathcal{W}[2]$ are $\mathcal{FPT}$-reducible to the problem, where $\mathcal{W}[2]$ is a parameterized complexity class with $\mathcal{W}[1] \subseteq \mathcal{W}[2]$.

	General Case	$k=1$	$k=2$	$k \geq 3$
adding votes	$r \leq 3: \mathcal{P}^{\diamondsuit}$ $r \geq 4$: $\mathcal{NP}$-c$^{\diamondsuit}$ <u>$\mathcal{FPT}$, Thm. 7</u>	$\mathcal{P}^{\spadesuit}$	r is not a constant: <u>$\mathcal{NP}$-c, Thm. 2</u> r is a constant: <u>$\mathcal{P}$, Thm. 1</u>	$r \leq 3: \mathcal{P}$ $r \geq 4$: <u>$\mathcal{NP}$-c, Thm. 5</u>
deleting votes	$r \leq 2$: $\mathcal{P}^{\diamondsuit}$ $r \geq 3$: $\mathcal{NP}$-c$^{\diamondsuit}$ <u>$\mathcal{FPT}$, Thm. 6</u>	$\mathcal{P}^{\spadesuit}$	$r \leq 2: \mathcal{P}$ $r \geq 3$: <u>$\mathcal{NP}$-c, Thm. 3</u>	$r \leq 2: \mathcal{P}$ $r \geq 3$: <u>$\mathcal{NP}$-c, Thm. 3</u>
adding candidates	$r=1$: $\mathcal{NP}$-c$^{\clubsuit}$	$\mathcal{P}^{\spadesuit}$	$r=1$: $\mathcal{NP}$-c$^{\triangle}$	$r=1$: $\mathcal{NP}$-c$^{\triangle}$
deleting candidates	$r=1$: $\mathcal{NP}$-c$^{\clubsuit}$	$\mathcal{P}^{\spadesuit}$	$r=1$: $\mathcal{NP}$-c$^{\triangle}$	$r=1$: <u>$\mathcal{W}[1]$-h, Thm. 8</u>

Table 1: A summary of the complexity of r-approval control problems. Our new results are highlighted with underlines. "Thm. #" means that the result follows from Theorem # in this paper. The two $\mathcal{FPT}$ results are with respect to the number of added/deleted votes as parameters, respectively. Results marked by $\diamondsuit$ are from [21], by $\clubsuit$ from [19], by $\spadesuit$ from [15] and by $\triangle$ from [14].

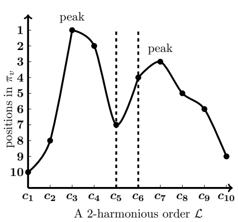

Figure 1: A 2-peaked vote $\pi_v = (c_3, c_4, c_7, c_6, c_8, c_9, c_5, c_2, c_{10}, c_1)$ with respect to $\mathcal{L} = (c_1, c_2, \ldots, c_{10})$. Here, $\mathcal{L}$ is partitioned into L_1 and L_2 with $L_1 = (c_1, c_2, c_3, c_4, c_5)$ and $L_2 = (c_6, c_7, c_8, c_9, c_{10})$.

$A \uplus B = \{1,1,1,1,2,2,3,3,3,4\}$ and $A \ominus B = \{1,1,3,4\}$. A multiset B is a *submultiset* of a multiset A, if, for every object s that occurs n times in B, A contains at least n copies of s. We use $B \sqsubseteq A$ to denote that B is a submultiset of A.

r-Approval. An r-approval election can be specified by a set $\mathcal{C}$ of candidates, a set $\mathcal{V}$ of voters where every $v \in \mathcal{V}$ casts a vote π_v which is defined as a linear order over the candidates $\mathcal{C}$. Each voter v contributes 1 point to each of the candidates ordered in the top r positions in π_v and contributes 0 points to the other candidates. For convenience, we also use $\succ_v$ to denote π_v. For a vote $\succ_v$ and a candidate c, the *position* of c in $\succ_v$ is defined as $|\{c' \in \mathcal{C} \mid c' \succ_v c\}|+1$, where $c' \succ_v c$ means that c' is ordered before c in $\succ_v$. The multiset of votes casted by $\mathcal{V}$ is denoted by $\Pi_{\mathcal{V}}$. A *winner* is a candidate who gets the highest total score. If there is only one winner, we call it a *unique winner*.

For simplicity, sometimes we use $(a_1, a_2, ..., a_n)$ to denote the linear order $a_1 \succ a_2 \succ, ..., \succ a_n$. For a vote π_v and a subset $C \subseteq \mathcal{C}$, let $\pi_v(C)$ denote the *partial vote* of π_v restricted to C such that in $\pi_v(C)$ every two distinct candidates in C preserve the same order as in π_v. For example, for a vote $\pi_v = (a,b,c,d,e)$, $\pi_v(\{b,d,e\}) = (b,d,e)$. For a multiset Π of votes and a subset $C \subseteq \mathcal{C}$, let $\Pi(C)$ be the multiset of votes obtained from Π by replacing each $\pi \in_+ \Pi$ by $\pi(C)$.

Single-peaked/k-peaked elections. An election $(\mathcal{C}, \Pi_{\mathcal{V}})$ is *single-peaked* if there is a linear order $\mathcal{L}$ of $\mathcal{C}$ such that for every $\succ_v$ in $\Pi_{\mathcal{V}}$ and every three candidates $a,b,c \in \mathcal{C}$ with $a \;\mathcal{L}\; b \;\mathcal{L}\; c$ or $c \;\mathcal{L}\; b \;\mathcal{L}\; a$, $c \succ_v b$ implies $b \succ_v a$, where $a \;\mathcal{L}\; b$ means a is ordered before b in $\mathcal{L}$. The candidate ordered in the first position of $\succ_v$ is the *peak* of $\succ_v$ with respect to $\mathcal{L}$.

For an order $\mathcal{L} = (c_1, c_2, \ldots, c_m)$ of $\mathcal{C}$ and a vote π_v, we say π_v is *k-peaked* with respect to $\mathcal{L}$, if there is a k-partition $L_1 = (c_1, c_2, ..., c_i)$, $L_2 = (c_{i+1}, c_{i+2}, ..., c_{i+j}), \ldots, L_k = (c_t, c_{t+1}, ..., c_m)$ of $\mathcal{L}$ such that $\pi_v(\mathcal{C}(L_i))$ is single-peaked with respect to L_i for all $1 \leq i \leq k$, where $\mathcal{C}(L_i)$ is the set of candidates appearing in L_i. See Fig. 1 for an example.

An election is k-peaked if there is an order $\mathcal{L}$ of $\mathcal{C}$ such that every vote in the election is k-peaked with respect to $\mathcal{L}$. Here $\mathcal{L}$ is called a *k-harmonious order.*

Problem definitions. The problems studied in this paper are defined as follows. Throughout this paper, we fix p as the distinguished candidate.

r-Approval Control by Adding Votes in k-Peaked Elections (r-AV-k)
Input: An r-approval election $(\mathcal{C} \cup \{p\}, \Pi_{\mathcal{V}})$ with a multiset $\Pi_{\mathcal{T}}$ of unregistered votes, where both $\Pi_{\mathcal{V}}$ and $\Pi_{\mathcal{T}}$ are k-peaked with respect to a given k-harmonious order $\mathcal{L}$, and an integer $0 \leq R \leq |\Pi_{\mathcal{T}}|$.
Question: Are there at most R votes $\Pi_{\mathcal{T}'}$ in $\Pi_{\mathcal{T}}$ such that p is the unique winner/a winner in $(\mathcal{C} \cup \{p\}, \Pi_{\mathcal{V}} \uplus \Pi_{\mathcal{T}'})$?

r-Approval Control by Deleting Votes in k-Peaked Elections (r-DV-k)
Input: A k-peaked r-approval election $(\mathcal{C} \cup \{p\}, \Pi_{\mathcal{V}})$ with a k-harmonious order $\mathcal{L}$, and an integer $0 \leq R \leq |\Pi_{\mathcal{V}}|$.
Question: Are there at most R votes $\Pi_{\mathcal{T}}$ in $\Pi_{\mathcal{V}}$ such that p is the unique winner/a winner in $(\mathcal{C} \cup \{p\}, \Pi_{\mathcal{V}} \ominus \Pi_{\mathcal{T}})$?

r-Approval Control by Deleting Candidates in k-Peaked Elections (r-DC-k)
Input: A k-peaked r-approval election $(\mathcal{C} \cup \{p\}, \Pi_{\mathcal{V}})$ with a k-harmonious order $\mathcal{L}$, and an integer $0 \leq R \leq |\mathcal{C}|$.
Question: Are there at most R candidates $C \subseteq \mathcal{C}$ such that p is the unique winner/a winner in the election $((\mathcal{C} \cup \{p\}) \setminus C, \Pi_{\mathcal{V}}((\mathcal{C} \cup \{p\}) \setminus C))$?

We use r-AV, r-DV and r-DC to denote the above problems without the restriction of k-peaked elections, respectively.

3. 2-PEAKED ELECTIONS

In this section, we study the control problems for r-approval in 2-peaked elections. The following three theorems summarize our findings.

THEOREM 1. *r-AV-2 is polynomial-time solvable for every constant r.*

Recall that r-AV is $\mathcal{NP}$-complete for every constant $r \geq 4$ but polynomial-time solvable when restricted to single-peaked elections [15]. Theorem 1 shows that the polynomial-time solvability of r-AV remains when extending from single-peaked elections to 2-peaked elections, for r being a constant. This bound is tight as indicated by the following theorem. More precisely, if r is not a constant, r-AV becomes $\mathcal{NP}$-complete in 2-peaked elections, in contrast to the polynomial-time solvability in the single-peaked case [15].

THEOREM 2. *r-AV-2 is $\mathcal{NP}$-complete, if r is not a constant.*

The control problem by deleting votes for r-approval is polynomial-time solvable in single-peaked elections for even non-constant r [15]. The following theorem shows that by increasing the number of peaks only by one, this problem becomes $\mathcal{NP}$-complete.

THEOREM 3. *r-DV-2 is $\mathcal{NP}$-complete for every constant $r \geq 3$.*

3.1 Proof of Theorem 1

We prove Theorem 1 by giving a polynomial-time algorithm based on dynamic programming.

Let $((\mathcal{C} \cup \{p\}, \Pi_{\mathcal{V}}), \Pi_{\mathcal{T}}, \mathcal{L}, R)$ be an instance of r-AV-2. For $c \in \mathcal{C}$, let $\overleftarrow{c}(1)$ be the candidate lying immediately before c in $\mathcal{L}$ and $\overleftarrow{c}(i)$ be the candidate lying immediately before $\overleftarrow{c}(i-1)$ in $\mathcal{L}$. Similarly, we use $\overrightarrow{c}(1)$ and $\overrightarrow{c}(i)$ to denote the candidates lying immediately after c and $\overrightarrow{c}(i-1)$, respectively. For example, if $\mathcal{L} = (a, b, c, d, e, f, g, h)$, then $\overrightarrow{d}(1) = e$, $\overrightarrow{d}(4) = h$, $\overleftarrow{d}(1) = c$ and $\overleftarrow{d}(3) = a$.

For a vote π_v, let $1(v)$ denote the set of candidates who get 1 point and $0(v)$ denote the set of candidates who get 0 points, from π_v. For a candidate c, let $SC_{\mathcal{V}}(c)$ be the total score of c from $\Pi_{\mathcal{V}}$, that is, $SC_{\mathcal{V}}(c) = |\{\pi_v \in_+ \Pi_{\mathcal{V}} \mid c \in 1(v)\}|$.

Given an order $A = (a_1, a_2, \ldots, a_n)$, a *discrete interval I* over A is a consecutive sub-order $(a_i, a_{i+1}, \ldots, a_{i+t})$ of A. We denote the first element a_i by $l(I)$ and the last element a_{i+t} by $r(I)$. We also use $A(l(I), r(I))$ to denote I. Let $\mathcal{S}(I)$ denote the set of elements appearing in I and set $|I| := |\mathcal{S}(I)|$. For example, for a discrete interval $I = A(3, 6)$ over the order $A = (2, 5, 3, 10, 4, 6, 0)$, $\mathcal{S}(I)$ is $\{3, 4, 6, 10\}$. A *k-discrete interval* over an order A is a collection of k disjoint discrete intervals over A, where "disjoint" means no element in A appears in more than one discrete interval. For a k-discrete interval $\mathcal{I}$, let $\mathcal{S}(\mathcal{I}) = \bigcup_{I \in \mathcal{I}} \mathcal{S}(I)$.

OBSERVATION 1. *For each k-peaked election $(\mathcal{C}, \Pi_{\mathcal{V}})$ associated with a k-harmonious order $\mathcal{L}$ over $\mathcal{C}$, and each vote $\pi_v \in_+ \Pi_{\mathcal{V}}$, there is a k'-discrete interval $\mathcal{I}$ over $\mathcal{L}$ such that $0 < k' \leq k$ and $1(v) = \mathcal{S}(\mathcal{I})$.*

By Observation 1, for every vote π_v in a 2-peaked election associated with $\mathcal{L}$ as a 2-harmonious order, $1(v)$ can be represented by a 2-discrete interval or a 1-discrete interval over $\mathcal{L}$. See Fig. 2 for an example.

We first derive a polynomial-time algorithm for 4-AV-2. It is easy to generalize the algorithm to r-AV-2 with r being a constant. The following observation is trivial.

OBSERVATION 2. *Every true-instance of r-AV has a solution where each added vote approves p.*

Due to Observation 2, we can safely assume that for each $\pi_v \in_+ \Pi_{\mathcal{T}}$, $p \in 1(v)$. By Observation 1, for every vote

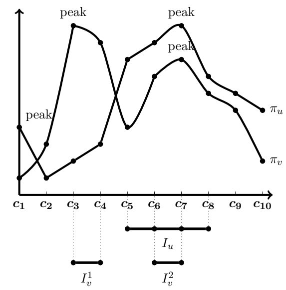

Figure 2: This figure shows two votes $\pi_v = (c_3, c_4, c_7, c_6, c_8, c_9, c_5, c_2, c_{10}, c_1)$ and $\pi_u = (c_7, c_6, c_5, c_8, c_9, c_{10}, c_1, c_4, c_3, c_2)$. Each vote gives one point to its top four ordered candidates. $1(v)$ is represented by a 2-discrete interval $\{I_v^1 = (c_3, c_4), I_v^2 = (c_6, c_7)\}$ and $1(u)$ is represented by a 1-discrete interval $\{I_u = (c_5, c_6, c_7, c_8)\}$.

$\pi_v \in_+ \Pi_{\mathcal{T}}$, $1(v)$ can be represented by a 2-discrete interval $\mathcal{I}_v = \{I_v^{\overline{p}}, I_v^p\}$ or a 1-discrete interval $\mathcal{I}_v = \{I_v^p\}$ with $p \in \mathcal{S}(I_v^p)$. Let S be the set of all votes $\pi_v \in_+ \Pi_{\mathcal{T}}$ where $1(v)$ is represented by a 1-discrete interval over $\mathcal{L}$. We say two votes have the same *type* if they approve the same candidates. Since every voter approves exactly four candidates, S has at most four different types: (1) votes approving $\overleftarrow{p}(3), \overleftarrow{p}(2), \overleftarrow{p}(1), p$; (2) votes approving $\overleftarrow{p}(2), \overleftarrow{p}(1), p, \overrightarrow{p}(1)$; (3) votes approving $\overleftarrow{p}(1), p, \overrightarrow{p}(1), \overrightarrow{p}(2)$; and (4) votes approving $p, \overrightarrow{p}(1), \overrightarrow{p}(2), \overrightarrow{p}(3)$. We then can try all possibilities of how many votes in the solution are from each of the four types of votes in S. This reduces the original instance to at most R^4 sub-instances. Thus, in the following, we assume that every vote in $\Pi_{\mathcal{T}}$ is represented by a 2-discrete interval. Let $\vec{\Pi}_{\mathcal{T}} = (\pi_{v_1}, \pi_{v_2}, \ldots, \pi_{v_{|\mathcal{T}|}})$ be an order of $\Pi_{\mathcal{T}}$ such that $r(I_{v_i}^{\overline{p}}) = r(I_{v_j}^{\overline{p}})$ or $r(I_{v_i}^{\overline{p}})\ \mathcal{L}\ r(I_{v_j}^{\overline{p}})$ for all $1 \leq i < j \leq |\Pi_{\mathcal{T}}|$.

Our dynamic programming algorithm uses a binary dynamic table $DT(i, j, s, s_1, s_2, s_3, s_4, s_5, s_6, s_{i,1}, s_{i,2}, s_{i,3})$, where we set $DT(i, j, s, s_1, s_2, s_3, s_4, s_5, s_6, s_{i,1}, s_{i,2}, s_{i,3}) = 1$ if there is a submultiset $\Pi_{\mathcal{T}'} \sqsubseteq \{\pi_{v_1}, \pi_{v_2}, \ldots, \pi_{v_i}\}$ satisfying

(1) $|\Pi_{\mathcal{T}'}| = j$;

(2) $\pi_{v_i} \in_+ \Pi_{\mathcal{T}'}$;

(3) $\max\{SC_{\mathcal{V} \cup \mathcal{T}'}(c) \mid c \in \mathcal{C}\} = s$;

(4) $SC_{\mathcal{V} \cup \mathcal{T}'}(c_t) = s_t$ for all $1 \leq t \leq 6$, where $c_3 = \overleftarrow{p}(1), c_2 = \overleftarrow{p}(2), c_1 = \overleftarrow{p}(3), c_4 = \overrightarrow{p}(1), c_5 = \overrightarrow{p}(2)$ and $c_6 = \overrightarrow{p}(3)$; and

(5) $SC_{\mathcal{V} \cup \mathcal{T}'}(c_{i,t}) = s_{i,t}$ for all $t \in \{1, 2, 3\}$, where $c_{i,1} = r(I_{v_i}^{\overline{p}}), c_{i,2} = \overleftarrow{c_{i,1}}(1)$ and $c_{i,3} = \overleftarrow{c_{i,1}}(2)$. (See Fig. 3 for an illustration of (4) and (5)).

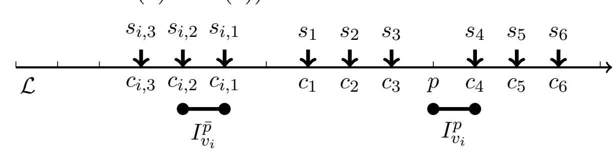

Figure 3: Illustration of (4) and (5) in the definition of DT.

It is easy to see that the given instance is a true-instance if there is a $DT(n, R', s, s_1, s_2, \ldots, s_6, s_{n,1}, s_{n,2}, s_{n,3}) = 1$ for some $n \leq |\Pi_{\mathcal{T}}|$, $R' \leq R$, $s \leq SC_{\mathcal{V}}(p) + R' - 1$ and $s' \leq s$ for all $s' \in \{s_1, s_2, \ldots, s_6, s_{n,1}, s_{n,2}, s_{n,3}\}$. Therefore, to solve the problem we need to calculate the values of $DT(i, j, s, s_1, s_2, \ldots, s_6, s_{i,1}, s_{i,2}, s_{i,3})$ for all $1 \leq j \leq R$, $j \leq i \leq |\Pi_{\mathcal{T}}|$, $1 \leq s \leq SC_{\mathcal{V}}(p) + R - 1$ and $s' \leq s$ for all $s' \in \{s_1, s_2, \ldots, s_6, s_{i,1}, s_{i,2}, s_{i,3}\}$. Thus, we have at most $|\mathcal{T}| \cdot R \cdot (|\mathcal{V}| + R)^{10}$ entries to calculate.

We use the following iterative recurrence to update the table. $DT(i, j, s, s_1, s_2, \ldots, s_6, s_{i,1}, s_{i,2}, s_{i,3}) = 1$, if at least one of the following cases applies:

Case 1. $\exists DT(i_1, j-1, s, s'_1, s'_2, \ldots, s'_6, s'_{i_1,1}, s'_{i_1,2}, s'_{i_1,3}) = 1$ such that conditions (1)-(4) hold.

Case 2. $\exists s' \in \{s_1, s_2, ..., s_6, s_{i,1}, s_{i,2}, s_{i,3}\}$ with $s' = s$ and $\exists DT(i_1, j-1, s-1, s'_1, s'_2, \ldots, s'_6, s'_{i_1,1}, s'_{i_1,2}, s'_{i_1,3}) = 1$ such that conditions (1)-(4) hold.

The four conditions are:

(1) $j - 1 \leq i_1 \leq i - 1$;

(2) $s_t = s'_t + SC_{\{v_i\}}(c_t)$ for all $1 \leq t \leq 6$;

(3) $s_{i,t} = s'_{i_1,t_1} + SC_{\{v_i\}}(c_{i,t})$ for all $c_{i,t} = c_{i_1,t_1}$; and

(4) $s_{i,t} = SC_{\mathcal{V}\cup\{v_i\}}(c_{i,t})$ for all $c_{i,t} \in \{r(I^{\bar{p}}_{v_i}), \overleftarrow{r(I^{\bar{p}}_{v_i})}(1), \overleftarrow{r(I^{\bar{p}}_{v_i})}(2)\} \setminus \{r(I^{\bar{p}}_{v_{i_1}}), \overleftarrow{r(I^{\bar{p}}_{v_{i_1}})}(1), \overleftarrow{r(I^{\bar{p}}_{v_{i_1}})}(2)\}$.

The algorithm is easy to be generalized for $r \geq 4$ by using a bigger but still polynomial-sized dynamic table.

3.2 Proof of Theorem 2

We prove Theorem 2 by a reduction from a variant of Independent Set which is $\mathcal{NP}$-complete [20].

Let ( ) denote an empty order containing no element. For a linear order $A = (a_1, a_2, \ldots, a_n)$, let $A[a_i, a_j]$ (resp. $A(a_i, a_j]$, $A[a_i, a_j)$ and $A(a_i, a_j)$) with $i \leq j$ be the sub-order $(a_i, a_{i+1}, \ldots, a_j)$ (resp. $(a_{i+1}, a_{i+2}, \ldots, a_j)$ if $i < j$ and ( ) if $i = j$, $(a_i, a_{i+1}, \ldots, a_{j-1})$ if $i < j$ and ( ) if $i = j$, and $(a_{i+1}, a_{i+2}, \ldots, a_{j-1})$ if $i < j-1$ and ( ) if $j \geq i \geq j-1$), and let $A[a_j, a_i]$ (resp. $A[a_j, a_i)$, $A(a_j, a_i]$ and $A(a_j, a_i)$) be the reversed order of $A[a_i, a_j]$ (resp. $A(a_i, a_j]$, $A[a_i, a_j)$ and $A(a_i, a_j)$). For two linear orders $A = (a_1, a_2, ..., a_n)$ and $B = (b_1, b_2, ..., b_m)$ with $A \cap B = \emptyset$, denote by (A, B) the linear order $(a_1, a_2, ..., a_n, b_1, b_2, ..., b_m)$. Let $[n]$ be the set $\{1, 2, ..., n\}$.

A Variant of Independent Set (VIS)

Input: A multiset $\mathcal{T} = \{T_1, T_2, ..., T_n\}$ where each $T_i \in_+ \mathcal{T}$ is a set of discrete intervals of size 4 over $(1, 2, ..., 12n)$ and $|T_i| \leq 3$ for all $T_i \in_+ \mathcal{T}$.

Question: Is there a set $S \subseteq \bigcup_{T\in_+\mathcal{T}} T$ of discrete intervals such that $|S| = n$, $|S \cap T_i| = 1$ for every $T_i \in_+ \mathcal{T}$ and no two discrete intervals in S intersect?

Given an instance $\mathcal{E} = (\mathcal{T} = \{T_1, T_2, \ldots, T_n\})$ of VIS, we construct an instance $\mathcal{E}' = ((\mathcal{C} \cup \{p\}, \Pi_\mathcal{V}), \Pi_\mathcal{T}, \mathcal{L}, R = n)$ for r-AV-2 as follows.

Let $\mathcal{I} = \bigcup_{T\in_+\mathcal{T}} T$. For each discrete interval $I \in \mathcal{I}$, let $l(I)$ be its left endpoint and $r(I)$ be its right endpoint. Let Γ be the set of all elements appearing in some discrete interval of $\mathcal{I}$, i.e., $\Gamma = \{\mathcal{S}(I) \mid I \in \mathcal{I}\}$. Let $\vec{\Gamma} = (x_1, x_2, \ldots, x_{|\Gamma|})$ be an order of Γ where $x_i < x_{i+1}$ for all $i \in [|\Gamma| - 1]$.

Candidates: We create three kinds of candidates C, D and E: (1) $C = \Gamma$; (2) D contains exactly $2n-1$ candidates $d_1, d_2, \ldots, d_n, \ldots, d_{2n-1}$; (3) E contains exactly $(n+3) \cdot (|C|+|D|-1)$ dummy candidates $x'_1, x'_2, ..., x'_{|C|\cdot(n+3)}, d'_1, d'_2, ..., d'_{(n+3)\cdot(|D|-1)}$ which will never be winners. The distinguished candidate is d_n, that is, $p = d_n$. Moreover, $r = n+4$.

2-Harmonious Order: $\mathcal{L} = (\vec{\Gamma}, \vec{D}, \vec{E})$ where $\vec{D} = (d_1, d_2, ..., d_{2n-1})$ and $\vec{E} = (x'_1, ..., x'_{|C|\cdot(n+3)}, d'_1, ..., d'_{(n+3)\cdot(|D|-1)})$

Registered Votes $\Pi_\mathcal{V}$**:** (1) for each $x_i \in C$, create $n-2$ votes defined as $(x_i, \mathcal{L}[x'_{(n+3)i-n-2}, x'_{i(n+3)}], \mathcal{L}(x_i, x_1], \mathcal{L}(x_i, x'_{(n+3)i-n-2}), \mathcal{L}(x'_{i(n+3)}, d'_{(|D|-1)\cdot(n+3)}])$; (2) for each $d_i \in D$ where $i \in [n-1]$, create $n-(i+1)$ votes defined as $(d_i, \mathcal{L}[d'_{(n+3)i-n-2}, d'_{i(n+3)}], \mathcal{L}(d_i, x_1], \mathcal{L}(d_i, d'_{(n+3)i-n-2}), \mathcal{L}(d'_{i(n+3)}, d'_{(|D|-1)\cdot(n+3)}])$; (3) for each $d_i \in D$ where $i \in \{n+1, n+2, \ldots, 2n-1\}$, create $i-(n+1)$ votes which is defined as $(d_i, \mathcal{L}[d'_{(n+3)i-2n-5}, d'_{(i-1)\cdot(n+3)}], \mathcal{L}(d_i, x_1], \mathcal{L}(d_i, d'_{(n+3)i-2n-5}), \mathcal{L}(d'_{(i-1)\cdot(n+3)}, d'_{(|D|-1)\cdot(n+3)}])$.

Unregistered Votes $\Pi_\mathcal{T}$**:** For each $I_{ij} \in T_i \in_+ \mathcal{T}$, create a corresponding unregistered vote which is defined as $(\mathcal{L}[l(I_{ij}), r(I_{ij})], \mathcal{L}[d_i, d'_{(|D|-1)\cdot(n+3)}], \mathcal{L}(l(I_{ij}), x_1], \mathcal{L}(r(I_{ij}), d_{i-1}])$. Clearly, this vote approves exactly all four candidates lying between $l(I_{ij})$ and $r(I_{ij})$ (including $l(I_{ij})$ and $r(I_{ij})$) in $\mathcal{L}$ and all candidates lying between d_i and d_{i+n-1} (including d_i and d_{i+n-1}) in $\mathcal{L}$. Thus, every unregistered vote approves d_n.

It is clear that all votes are 2-peaked with respect to $\mathcal{L}$. Due to the construction, it is easy to see that $SC_\mathcal{V}(c) = n-2$ for all $c \in C$, $SC_\mathcal{V}(d_i) = n-i-1$ for all $d_i \in D$ with $i \in [n-1]$, $SC_\mathcal{V}(d_i) = i-n-1$ for all $d_i \in D$ with $i \in \{n+1, n+2, \ldots, 2n-1\}$, and $SC_\mathcal{V}(c) \leq n-2$ for all $c \in E$ and $SC_\mathcal{V}(d_n) = 0$.

$\Rightarrow$: Suppose that $\mathcal{E}$ is a true-instance and let S be a solution for $\mathcal{E}$. Let $\vec{S} = (I_1, I_2, \ldots, I_n)$ be an order of S where $I_i = S \cap T_i$ for all $i \in [n]$. Then, we can make d_n the unique winner by adding votes from $\Pi_\mathcal{T}$ according to S. More specifically, for each $I_i \in S$ we select its corresponding vote constructed as above and add it to the election. Clearly, the final score of d_n is n. Due to the construction, no two added votes π_v and π_u which correspond to two different intervals I_i and I_j, respectively, approve a common candidate from C. Thus, after adding these votes to the election, no candidate in C has a higher score than that of d_n. To analyze the score of $d_j \in D$ with $j \in [n-1]$, we observe that for any $i > j$ the vote corresponding to I_i does not approve d_j. Since $SC_\mathcal{V}(d_j) = n-j-1$ and $|S \cap T_i| = 1$ for all $i \in [j]$, we know that the final score of d_j is less than n. Similarly, to analyze the score of $d_j \in D$ with $j \in \{n+1, n+2, \ldots, 2n-1\}$, we observe that for any $i \leq j-n$ the vote corresponding to I_i does not approve d_j. Since $SC_\mathcal{V}(d_j) = j-n-1$ and $|S \cap T_i| = 1$ for all $i \in \{j-n+1, j-n+2, ..., n\}$, we know that the final score of d_j is less than n. The final score of each $c \in E$ is clearly less than $n-2$ since no unregistered vote approves c. Summarize the above analysis, we conclude that the distinguished candidate d_n becomes the unique winner after adding the selected votes to the election.

$\Leftarrow$: Suppose that $\mathcal{E}'$ is a true-instance and S' is a multiset of votes chosen from $\Pi_\mathcal{T}$ which makes d_n the unique winner in the election $(C \cup D \cup E, \Pi_\mathcal{V} \uplus S')$. It is easy to verify that $|S'| = n$, since otherwise, at least one of C would be a winner; thus, the final score of d_n is n and every $c \in C$ can get at most one point from S'. Therefore, no two votes in S' approve a common candidate of C, implying that S' must be a set. Let $P_1, P_2, \ldots, P_n$ be a partition of $\Pi_\mathcal{T}$ where P_i contains all votes corresponding to the intervals of $T_i \in_+ \mathcal{T}$. Clearly, P_i is a set. We claim here that $|S' \cap P_i| = 1$ for every $i \in [n]$. Suppose this is not true, then there must be a certain P_i with $|S' \cap P_i| \geq 2$. Let $S_1 = S' \cap P_i$ (thus, $|S_1| \geq 2$), $S_2 = \{\pi_v \in S' \cap P_{i'} \mid i' < i\}$ and $S_3 = \{\pi_v \in S' \cap P_{i'} \mid i' > i\}$. It is clear that $|S_1| + |S_2| + |S_3| = n$. Since all votes in S_1 approve both d_i and d_{i+n-1}, all votes in S_2 approve d_i but do not approve d_{i+n-1}, and all votes in S_3 approve d_{i+n-1} but do not approve d_i, then,

$$
\begin{aligned}
& SC_{\mathcal{V}\uplus S'}(d_i) + SC_{\mathcal{V}\uplus S'}(d_{i+n-1}) \\
&= SC_\mathcal{V}(d_i) + |S_1| + |S_2| + SC_\mathcal{V}(d_{i+n-1}) + |S_1| + |S_3| \\
&= n - i - 1 + |S_1| + |S_2| + i - 2 + |S_1| + |S_3| \\
&= 2n - 3 + |S_1| \\
&\geq 2n - 1
\end{aligned}
$$

Thus, at least one of d_i and d_{i+n-1} has final score at least n, contradicting that d_n is the unique winner. The claim is true. It is now easy to see that the set of discrete intervals corresponding to S' forms a solution for $\mathcal{E}$.

3.3 Proof of Theorem 3

We first prove that 3-DV-2 is $\mathcal{NP}$-hard by a reduction from VERTEX COVER on bounded degree-3 graphs which is $\mathcal{NP}$-complete [17]. Then, we will show that the proof applies to r-DV-2 for $r \geq 4$ with a slight modification.

An *undirected graph* is a tuple $G = (V, E)$ where V is the set of vertices and E is the set of edges. We also use $V(G)$ to denote the vertex set of G. For a vertex $u \in V$, $N_G(u)$ denotes the set of its *neighbors* in G, that is, $N_G(u) = \{w \mid (w, u) \in E\}$. The *degree* of a vertex u is the number of its neighbors. A graph is a *bounded degree-3 graph* if it contains at least one degree-3 vertex but no vertex having degree greater than 3. A *vertex cover* for a graph $G = (V, E)$ is a subset $S \subseteq V$ such that every edge in E has at least one of its endpoints in S.

Vertex Cover on Bounded Degree-3 Graphs (VC3)
Input: A bounded degree-3 graph $G = (V, E)$ and a positive integer κ.
Question: Does G have a vertex cover of size at most κ?

To prove the $\mathcal{NP}$-hardness of 3-DV-2, we first introduce a property for bounded degree-3 graphs. This property may be of independent interest since many graph problems are $\mathcal{NP}$-hard when restricted to graphs with bounded degree 3.

An *interval* over the real line is a closed set $[a, b] = \{x \in \mathbb{R} \mid a \leq x \leq b\}$ where a and b are real numbers. An interval is *trivial* if $a = b$. For an interval I, denote by $l(I)$ and $r(I)$ its left-point and right-point, respectively. A *t-interval* is a set of t intervals over the real line. A graph $G = (V, E)$ is a *t-interval graph* if there is a set $\mathcal{T}_G$ of t-intervals and a bijection $f : V \to \mathcal{T}_G$ such that for every $u, w \in V$, $(u, w) \in E$ if and only if $f(u)$ and $f(w)$ intersect. Here, $\mathcal{T}_G$ is called a *t-interval representation* of G. For simplicity, we use $\mathcal{I}_u = \{I_u^1, I_u^2, ..., I_u^t\}$ to denote $f(u)$. For two real numbers a and b with $a \leq b$, we define $(a, b) = \{x \in \mathbb{R} \mid a < x < b\}$.

The following lemma states that every bounded degree-3 graph has a 2-interval representation such that every vertex is represented by a 2-interval with one interval is trivial, and two 2-intervals only intersect at the endpoints.

LEMMA 4. *For every bounded degree-3 graph G there is a 2-interval representation for G such that for every $u \in V(G)$, $\mathcal{I}_u = \{I_u^1, I_u^2\}$ satisfies one of the following:*

1. $I_u^1 = [x_1, x_1]$, $I_u^2 = [x_2, x_3]$, $x_1 < x_2 < x_3$ *and* $\nexists u' \in V(G) \setminus \{u\}$ *such that* $r(I(u')) \in (x_2, x_3)$ *or* $l(I(u')) \in (x_2, x_3)$;
2. $I_u^1 = [x_1, x_2]$, $I_u^2 = [x_3, x_3]$, $x_1 < x_2 < x_3$ *and* $\nexists u' \in V(G) \setminus \{u\}$ *such that* $r(I(u')) \in (x_1, x_2)$ *or* $l(I(u')) \in (x_1, x_2)$,

for each $I(u') \in \{I_{u'}^1, I_{u'}^2\}$. Moreover, such a 2-interval representation can be found in polynomial time. See Fig. 4 for an example.

We now show the reduction. Let $\mathcal{E} = (G, \kappa)$ be an instance of VC3 and $\mathcal{I}(G)$ be a 2-interval representation of G satisfying all conditions in Lemma 4. For every $\mathcal{I}_u = \{I_u^1, I_u^2\}$, let $D(u)$ be the endpoints of I_u^1 and I_u^2 (due to

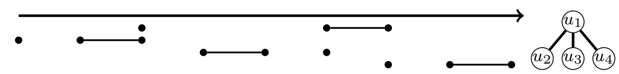

Figure 4: The left-hand figure illustrates a 2-interval representation of the right-hand graph. Here, the 2-intervals from up to down represent the vertices u_1, u_2, u_3 and u_4, respectively.

Lemma 4, $|D(u)| = 3$ for all $u \in V(G)$), and let $\Gamma = \bigcup_{u \in V(G)} D(u)$. Let $\vec{\Gamma} = (x_1, x_2, \ldots, x_{|\Gamma|})$ be the order of Γ with $x_i < x_{i+1}$ for all $i \in [|\Gamma| - 1]$. We construct an instance $\mathcal{E}' = ((\mathcal{C} \cup \{p\}, \Pi_{\mathcal{V}}), R = \kappa, \mathcal{L})$ of 3-DV-2 as follows.

Candidates: $\mathcal{C} = \Gamma \cup \{p, c_1, c_2, c_3, c_4\}$ with c_1, c_2, c_3, c_4 being dummy candidates, which would never be winners.

2-Harmonious Order: $\mathcal{L} = (\vec{\Gamma}, p, c_1, c_2, c_3, c_4)$.

Votes: There are two types of votes: votes disapproving p and votes approving p. There are $|V(G)|$ votes of the first type each of which corresponds to an $\mathcal{I}_u$ in $\mathcal{I}(G)$ for $u \in V(G)$. More specifically, for every $\mathcal{I}_u$, let (x_i, x_j, x_k) be the order of $D(u)$ with $x_i < x_j < x_k$, then we create a vote $\pi_u = (x_i, x_j, x_k, \mathcal{L}(x_i, x_1], \mathcal{L}(x_i, x_j), \mathcal{L}(x_j, x_k), \mathcal{L}(x_k, c_4])$. Thus, π_u approves $D(u)$. Due to Lemma 4, either x_i or x_k lies consecutively with x_j in $\mathcal{L}$, that is, one of $x_i = \overleftarrow{x_j}(1)$ and $x_k = \overrightarrow{x_j}(1)$ must hold, which implies that all votes of the first type are 2-peaked with respect to $\mathcal{L}$. There are only two votes of the second type: $(p, c_1, c_2, c_3, c_4, \mathcal{L}(p, x_1])$ and $(p, c_3, c_4, c_1, c_2, \mathcal{L}(p, x_1])$. It is clear that these two votes are 2-peaked with respect to $\mathcal{L}$.

In the following, we prove that $\mathcal{E}$ is a true-instance if and only if $\mathcal{E}'$ is a true-instance.

($\Rightarrow$:) Suppose that $\mathcal{E}$ is a true-instance and S is a vertex cover of size at most κ of G. Then, we delete all votes in $\{\pi_u \mid u \in S\}$. After deleting these votes, no two votes of the first type approve a common candidate in $\mathcal{C}$, since otherwise, $V(G) \setminus S$ could not be an independent set, contradicting the fact that S is a vertex cover. Thus, after deleting these votes all candidates except for p have only one point. Since p has two points, p is the unique winner.

($\Leftarrow$:) Suppose that $\mathcal{E}'$ is a true-instance. Observe that every true-instance of 3-DV has a solution containing only votes which do not approve p. Let S' be such a solution of size at most κ. Therefore, p has two points in the election after removing all votes in S'; thus every other candidate can have at most one point after removing all votes of S', which implies that no two votes of the first type approve a common candidate in $\mathcal{C}$ in the final election, further implying that the vertices corresponding to S' form a vertex cover of size at most κ for G.

In order to prove that r-DV-2 is $\mathcal{NP}$-hard for any constant $r \geq 4$, we need to modify the proof slightly. First, we add some dummy candidates. More specifically, there are $t = r - 3$ dummy candidates $X_i = \{x_i^1, x_i^2, ..., x_i^t\}$ with the order $(x_i^1, x_i^2, ..., x_i^t)$ between $x_i \in \Gamma$ and $x_{i+1} \in \Gamma$ in the 2-harmonious order $\mathcal{L}$ whenever there is a $u \in V(G)$ such that $[x_i, x_{i+1}] \in \mathcal{I}_u$. Besides, we have other $2r - 6$ dummy candidates $c_5, c_6, ..., c_{2r-2}$ lying after c_4 in $\mathcal{L}$, with the order $(c_5, c_6, ..., c_{2r-2})$. Thus, there are totally $t \cdot |V(G)| + 2r - 6$ new dummy candidates here. We change the first type of votes as follows: for every $u \in V(G)$ with $\mathcal{I}_u = \{[x_i, x_{i+1}], [x_j, x_j]\}$ (resp. $\mathcal{I}_u = \{[x_i, x_i], [x_j, x_{j+1}]\}$), we create a vote defined as $(\mathcal{L}[x_i, x_{i+1}], x_j, \mathcal{L}(x_i, x_1], \mathcal{L}(x_{i+1}, x_j), \mathcal{L}(x_j, c_{2r-2}])$ (resp. $(x_i, \mathcal{L}[x_j, x_{j+1}], \mathcal{L}(x_i, x_1], \mathcal{L}(x_i, x_j),$

$\mathcal{L}(x_{j+1}, c_{2r-2}])$). As for the second type of votes, we have still two votes defined as $(\mathcal{L}[p, c_{2r-2}], \mathcal{L}(p, x_1])$ and $(p, \mathcal{L}[c_r, c_{2r-2}], \mathcal{L}[c_1, c_r), \mathcal{L}(p, x_1])$, respectively. Then, with the same argument, we can show that r-DV-2 is $\mathcal{NP}$-hard for any $r \geq 4$.

4. 3-PEAKED ELECTIONS

In Section 2, we proved that control by adding votes in r-approval is polynomial-time solvable when restricted to 2-peaked elections and r being a constant. In this section, we show that the tractability of the problem does not hold when extended to 3-peaked elections.

THEOREM 5. *r-AV-3 is $\mathcal{NP}$-complete for every constant $r \geq 4$.*

PROOF. We first prove the $\mathcal{NP}$-hardness of 4-AV-3 by a reduction from INDEPENDENT SET on bounded degree-3 graphs which is $\mathcal{NP}$-complete [17]. An *independent set* in a graph $G = (V, E)$ is a subset $S \subseteq V$ such that every edge in E has at most one of its endpoints in S.

Independent Set on Bounded Degree-3 graphs (IS3)
Input: A bounded degree-3 graph $G = (V, E)$ and a positive integer κ.
Question: Does G have an independent set containing exactly κ vertices?

For an instance $\mathcal{E} = (G, \kappa)$ of IS3, let $\mathcal{I}(G)$ be a 2-interval representation of G which satisfies all conditions in Lemma 4. Let $D(u), \Gamma$ and $\vec{\Gamma}$ be defined as in Subsect. 3.3. We construct an instance $\mathcal{E}' = ((\mathcal{C} \cup \{p\}, \Pi_\mathcal{V}), \Pi_\mathcal{T}, \mathcal{L}, R = \kappa)$ of 4-AV-3 as follows.

Candidates: $\mathcal{C} = \Gamma \cup \{p, c_1, c_2, c_3\}$.

3-Harmonious Order: $\mathcal{L} = (\vec{\Gamma}, p, c_1, c_2, c_3)$.

Registered Votes $\Pi_\mathcal{V}$**:** The role of registered votes is to guarantee that all candidates of Γ have the same score $\kappa - 2$. To this end, we first create $\kappa - 2$ votes defined as $(\mathcal{L}[x_i, x_{i+3}], \mathcal{L}(x_i, x_1], \mathcal{L}(x_{i+3}, c_3])$ for every $i = 1, 5, \ldots, 4\lfloor|\Gamma|/4\rfloor - 3$. Then, we create some further votes according to $|\Gamma|$. **Case 1.** $|\Gamma| \equiv 0 \mod 4$. We create no further vote. **Case 2.** $|\Gamma| \equiv 1 \mod 4$. We create additional $\kappa - 2$ votes defined as $(x_{|\Gamma|}, \mathcal{L}[c_1, c_3], \mathcal{L}(x_{|\Gamma|}, x_1], p)$. **Case 3.** $|\Gamma| \equiv 2 \mod 4$. We create additional $\kappa - 2$ votes defined as $(x_{|\Gamma|-1}, x_{|\Gamma|}, c_1, c_2, \mathcal{L}(x_{|\Gamma|-1}, x_1], p, c_3)$. **Case 4.** $|\Gamma| \equiv 3 \mod 4$. We create additional $\kappa - 2$ votes defined as $(\mathcal{L}[x_{|\Gamma|-2}, x_{|\Gamma|}], c_1, \mathcal{L}(x_{|\Gamma|-2}, x_1], p, c_2, c_3)$.

Unregistered Votes $\Pi_\mathcal{T}$**:** For each $u \in V(G)$, let (x_i, x_j, x_k) be the order of $D(u)$ with $x_i < x_j < x_k$. We create a vote $\pi_u = (x_i, x_j, x_k, p, \mathcal{L}(x_i, x_1], \mathcal{L}(x_i, x_j), \mathcal{L}(x_j, x_k), \mathcal{L}(x_k, p), \mathcal{L}(p, c_3])$. Due to Lemma 4, either x_i or x_k lies consecutively with x_j in $\mathcal{L}$; thus, all these unregistered votes have 3 peaks x_α, x_β and p where $\{x_\alpha, x_\beta\} \subseteq \{x_i, x_j, x_k\}$ ($\{x_\alpha, x_\beta\}$ depends on whether x_j lies consecutively with x_i or with x_k), with respect to $\mathcal{L}$.

In the following, we prove that $\mathcal{E}$ is a true-instance if and only if $\mathcal{E}'$ is a true-instance. It is easy to see that $SC_\mathcal{V}(x) = \kappa - 2$ for all $x \in \mathcal{C} \setminus \{p, c_1, c_2, c_3\}$, $SC_\mathcal{V}(p) = 0$ and $SC_\mathcal{V}(c) \leq \kappa - 2$ for all $c \in \{c_1, c_2, c_3\}$.

($\Rightarrow$:) Suppose that $\mathcal{E}$ is a true-instance and S is an independent set of size κ. Then we add all votes corresponding to S, that is, all votes in $\{\pi_u \mid u \in S\}$, to the election. Since S is an independent set, no two added votes approve a common candidate except p; thus, each candidate except p has a final score at most $\kappa - 1$. Since each added vote approves p, it follows that p has a final score of κ points, implying that p becomes the unique winner in the election including the new votes.

($\Leftarrow$:) Suppose that $\mathcal{E}'$ is a true-instance and S' is a solution. Clearly, p has a final score of κ points. Since p is the unique winner, for every $c \in \mathcal{C} \setminus \{p\}$, there is at most one vote in S' approving c. Thus, no two votes in S' approve a common candidate except p. Due to the construction, the vertices corresponding to S' must be an independent set.

The proof applies to r-AV-3 for any constant $r \geq 5$ by a similar modification as discussed in Subsect. 3.3. □

5. PARAMETERIZED COMPLEXITY

In this section, we study several control problems for r-approval from the viewpoint of parameterized complexity. The first two $\mathcal{FPT}$ results are for the general case.

THEOREM 6. *r-DV is $\mathcal{FPT}$ with the number of deleted votes as the parameter, where r is a constant.*

PROOF. To derive the $\mathcal{FPT}$ algorithm, we divide the candidates $\mathcal{C}$ into two parts: $\mathcal{C}_1 = \{c \in \mathcal{C} \mid SC_\mathcal{V}(c) \geq SC_\mathcal{V}(p)\}$ and $\mathcal{C}_2 = \mathcal{C} \setminus \mathcal{C}_1$. Meanwhile, we divide $\Pi_\mathcal{V}$ into two parts: $\Pi_{\mathcal{V}_1} = \{\pi_v \in_+ \Pi_\mathcal{V} \mid \exists c \in \mathcal{C}_1 \text{ with } c \in 1(v)\}$ and $\Pi_{\mathcal{V}_2} = \Pi_\mathcal{V} \ominus \Pi_{\mathcal{V}_1}$. That is, $\Pi_{\mathcal{V}_1}$ contains all the votes which approve at least one candidate which has at least the same score as p, and $\Pi_{\mathcal{V}_2}$ contains other votes. We observe that every true-instance of r-DV has a solution S with $S \sqsubseteq \Pi_{\mathcal{V}_1}$. Due to the observation, we can restrict our attention to $\Pi_{\mathcal{V}_1}$. Since at most R votes can be deleted and each deleted vote approves at most r candidates, $|\mathcal{C}_1| \leq r \cdot R$ holds for every true-instance. We assume now that $|\mathcal{C}_1| \leq r \cdot R$ in the given instance. We say two votes have the same *type* if they approve the same candidates. Clearly, there are at most $O((r \cdot R)^r)$ different types of votes in $\Pi_{\mathcal{V}_1}$. Since every solution includes at most R votes from each type of votes, we have at most $R^{O(f(R))}$ cases to check where $f(R) = R^r$, implying an $\mathcal{FPT}$ algorithm for r-DV. □

THEOREM 7. *r-AV is $\mathcal{FPT}$ with the number of added votes as the parameter, where r is a constant.*

Faliszewski et al. [14] proved that control by deleting candidates in 1-approval is $\mathcal{NP}$-hard when restricted to Swoon-SP elections. Since Swoop-SP is a special case of 2-peaked elections, 1-DC-k with $k \geq 2$ is $\mathcal{NP}$-hard. We strengthen this result by proving that 1-DC-3 is $\mathcal{W}[1]$-hard with the number of deleted candidates as the parameter.

THEOREM 8. *1-DC-3 is $\mathcal{W}$[1]-hard with the number of deleted candidates as the parameter.*

PROOF. We prove the theorem by an $\mathcal{FPT}$ reduction from INDEPENDENT SET which is $\mathcal{W}[1]$-hard [10]. For a linear order $\vec{A} = (a_1, a_2, ..., a_n)$ over $A = \{a_1, a_2, ..., a_n\}$ and a subset $B \subseteq A$, denote by $\vec{A} \setminus B$ the linear order of $A \setminus B$ obtained from $\vec{A}$ by deleting all elements in B. For an instance $\mathcal{E} = (G = (V, E), \kappa)$ of Independent Set we construct an instance $\mathcal{E}'$ of 1-DC-3 as follows.

Candidates: $V \cup \{p, a, a_1, a_2, ..., a_\kappa, b, b_1, b_2, ..., b_\kappa\}$.

3-Harmonious Order: Let $\mathcal{F} = (c_1, c_2, ..., c_n)$ be an (arbitrary) order of V. Then, the 3-harmonious order $\mathcal{L}$ is given by $(b_\kappa, b_{\kappa-1}, ..., b_1, b, p, a, a_1, a_2, ..., a_\kappa, c_1, c_2, ..., c_n)$.

Votes: Let $m := |E|$. There are seven types of votes. (1) $2m - 1$ votes defined as $(\mathcal{L}[a, c_n], \mathcal{L}[p, b_\kappa])$; (2) $2m$ votes defined as $(\mathcal{L}[p, c_n], \mathcal{L}[b, b_\kappa])$; (3) $2m + \kappa - 1$ votes defined as $(\mathcal{L}[b, b_\kappa], \mathcal{L}[p, c_n])$; (4) for each edge $\{c_i, c_j\} \in E(G)$ with $i < j$, create one vote defined as $(c_i, c_j, \mathcal{L}[a, a_\kappa], \mathcal{L}[p, b_\kappa], \mathcal{F} \setminus \{c_i, c_j\})$; (5) for each vertex c_i, create one vote defined as $(c_i, \mathcal{L}[p, a_\kappa], \mathcal{L}[b, b_\kappa], \mathcal{F} \setminus \{c_i\})$ and one vote defined as $(c_i, \mathcal{L}[a, a_\kappa], \mathcal{L}[p, b_\kappa], \mathcal{F} \setminus \{c_i\})$; (6) $\kappa + 1$ votes defined as $(\mathcal{L}[a_1, c_n], \mathcal{L}[a, b_\kappa])$; (7) one vote defined as $(\mathcal{L}[b_1, b_\kappa], \mathcal{L}[b, c_n])$; It is easy to verify that all constructed votes are 3-peaked with respect to $\mathcal{L}$.

Number of Added Candidates: $R = \kappa$.

($\Leftarrow$:) It is easy to verify that $\mathcal{E}$ is a true-instance implies $\mathcal{E}'$ is a true-instance: for every independent set S of size κ, deleting the candidates S from the election clearly make the distinguished candidate p become the unique winner.

($\Rightarrow$:) Suppose that $\mathcal{E}'$ is a true-instance and S' is a solution with $|S'| \leq \kappa$. We first observe that $b \notin S'$. This observation is true, since otherwise, all candidates in $\{b_1, b_2, ..., b_\kappa\}$ must be deleted, contradicting that $|S'| \leq \kappa$. The same argument applies to the candidate a. However, in order to beat b, exactly k candidates from V must be deleted so that p can get extra k points from the constructed votes of case 5. Since $|S'| \leq \kappa$, S' must be a subset of V. Moreover, no two candidates $c_1, c_2 \in S'$ are adjacent to each other in the graph G, since otherwise, the candidate a would get at least one extra point from the constructed votes of case 4, and p cannot be the unique winner. Thus, S' forms an independent set of size κ of G. $\square$

6. CONCLUSION

In this paper, we study the k-peaked elections which generalize the single-peaked elections by allowing at most k-peaks in each vote. We derive a dichotomy of the complexity of control problems for r-approval voting in k-peaked elections with respect to k. Moreover, we present some results concerning the parameterized complexity of these problems in general as well as k-peaked elections. All of our results work for both unique-winner and nonunique-winner models. In addition, several of our results apply to approval voting and SP-AV as well. Another possible research direction could be studying more strategic behaviors for other voting systems in k-peaked elections.

Acknowledgments

We thank the AAMAS-14 reviewers for their helpful comments. Yongjie Yang was supported by the DFG Cluster of Excellence (MMCI) and the China Scholarship Council (CSC). Jiong Guo was supported by the DFG Cluster of Excellence (MMCI).

7. REFERENCES

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

[2] N. Betzler, R. Bredereck, J. Chen, and R. Niedermeier. Studies in computational aspects of voting - a parameterized complexity perspective. In *The Multivariate Algorithmic Revolution and Beyond*, pages 318–363, 2012.

[3] N. Betzler and J. Uhlmann. Parameterized complexity of candidate control in elections and related digraph problems. *Theor. Comput. Sci.*, 410(52):5425–5442, 2009.

[4] D. Black. On the rationable of group decition-making. *Journal of Political Economy*, 56:23–34, 1948.

[5] S. Brams and M. R. Sanver. Critical strategies under approval voting: Who gets ruled in and ruled out. *Electoral Studies*, 25(2):287–305, 2006.

[6] F. Brandt, M. Brill, E. Hemaspaandra, and L. A. Hemaspaandra. Bypassing combinatorial protections: Polynomial-time algorithms for single-peaked electorates. In *AAAI*, pages 715–722, 2010.

[7] R. Bredereck, J. Chen, and G. J. Woeginger. Are there any nicely structured preference profiles nearby? In *IJCAI*, 2013.

[8] D. Cornaz, L. Galand, and O. Spanjaard. Kemeny elections with bounded single-peaked or single-crossing width. In *IJCAI*, 2013.

[9] G. Demange. Single-peaked orders on a tree. *Mathematical Social Sciences*, 3(3):389–396, 1983.

[10] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer, 1999.

[11] G. Erdélyi, M. Lackner, and A. Pfandler. Computational aspects of nearly single-peaked electorates. In *AAAI*, 2013.

[12] G. Erdélyi, M. Nowak, and J. Rothe. Sincere-strategy preference-based approval voting fully resists constructive control and broadly resists destructive control. *Math. Log. Q.*, 55(4):425–443, 2009.

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

[14] P. Faliszewski, E. Hemaspaandra, and L. A. Hemaspaandra. The complexity of manipulative attacks in nearly single-peaked electorates. In *TARK*, pages 228–237, 2011.

[15] 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.

[16] S. Gailmard, J. W.Patty, and E. Maggie Penn. Arrow's theorem on single-peaked domains. *The Political Economy of Democoracy*, pages 235–342, 2009.

[17] M. R. Garey and D. S. Johnson. The rectilinear steiner tree problem is NP-complete. *SIAM Journal of Applied Mathematics*, 32:826–834, 1977.

[18] 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.

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

[20] J. Mark Keil. On the complexity of scheduling tasks with discrete starting times. *Operations Research Letters*, 12(5):293–295, 1992.

[21] A. Lin. The complexity of manipulating k-approval elections. In *ICAART (2)*, pages 212–218, 2011.

[22] H. Liu, H. Feng, D. Zhu, and J. Luan. Parameterized computational complexity of control problems in voting systems. *Theor. Comput. Sci.*, 410(27-29):2746–2753, 2009.

[23] T. Walsh. Uncertainty in preference elicitation and aggregation. In *AAAI*, pages 3–8, 2007.