Abstract. Single-peaked elections have been attracting much attention recently. It turned out that many \(NP\)-hard voting problems become polynomial-time solvable when restricted to single-peaked elections. A natural generalization of the single-peaked elections is the \(k\)-peaked elections, where at most \(k\) peaks are allowed in each vote in the election. In this paper, we mainly aim at establishing a complexity dichotomy of controlling behaviors of a variant of the sincere-strategy preference-based approval voting in \(k\)-peaked elections for different values of \(k\). It turns out that most \(NP\)-hardness results in the general case also hold in \(k\)-peaked elections, even for \(k = 2, 3\). On the other hand, we derive polynomial-time algorithms for certain sincere-strategy preference-based approval voting control problems for \(k = 2\). In addition, we also study the sincere-strategy preference-based approval control problems from the viewpoint of parameterized complexity and prove some \(FPT\) and \(W\)-hardness results.

1 Introduction

Voting has been a common method to get aggregation results from different preferences. It has wide applications in practice, such as political election, multi-agent system, web spam reduction etc. However, by Arrow’s impossibility theorem [1], there is no natural voting system satisfying a set of reasonable criteria at one time when the voting involves at least three candidates. One way to avoid Arrow’s impossibility theorem is to constraint the domain of aggregation rules. Single-peaked model, which was introduced by Black [6] in 1948, has been one of the most important restrictions. Intuitively, the voter are restricted in a way that there exists an ordering of candidates where each vote has an ideal candidate and the candidates that are farther from the ideal candidate is preferred less. In the single-peaked model, the majority rule meet all the reasonable criteria stated in the Arrow’s impossibility theorem. Due to its importance, the single-peaked model has been extensively studied in lots of literature. Recently, the complexity of many voting problems in single-peaked elections has been attracting much attention [8, 16, 5, 27]. It turned out that many voting problems being \(NP\)-hard in general become polynomial-time solvable when restricted to single-peaked elections. However, many elections in practice are not purely single-peaked, which motivates people to study more general models of elections. We refer readers to [15, 12, 10] for some variations of single-peaked model.

In this paper, we consider a natural generalization of single-peaked model, where more than one peak may occur in each vote. This model arises in some practical settings, especially in some elections involving an axis over all candidates. 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 some special days which he/she prefers for some reason, and the farther the other day lies away from these favorites days, the less it is preferred by the voter. We call such a model \(k\)-peaked elections where at most \(k\) peaks are allowed in each vote with respect to a given ordering over all candidates. Thus, 1-peaked elections are exactly the single-peaked elections, while \([m/2]\)-peaked elections become the general case, where \(m\) is the number of candidates.

Certain issues which have been attracting much attention in voting systems are the strategic attacks. Controlling behaviors are well-known and extensively studied strategic attacks. In a controlling 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 want to reach. One goal is to make some distinguished candidate win the election. The other goal could be to make someone lose the election. We call the former case a constructive control and the latter one a destructive control. On the other hand, the tricks involved in a controlling attack include adding some new unregistered votes to the registered votes, deleting votes from the registered votes, adding new
Approval voting system is one of the most famous voting systems which has been extensively studied both in theory and in practice. In an approval voting (AV) system, we are given a set $C$ of candidates and a set $V$ of voters. Each voter approves or disapproves every candidate $c \in C$. The system selects a candidate who is approved by the most voters as a winner. A shortcoming of AV is its vulnerability to many types of controlling attacks [3]. The reason is that in an AV election each voter considers only approving or disapproving a candidate but ignores the preference over the candidates. To overcome this shortcoming, Brams and Sanver [7] proposed a variant of AV which is commonly called “sincere-strategy preference-based approval voting (SP-AV for short). In an SP-AV election, each voter $v$ casts a vote $\pi_v$ which is defined as a linear order (a transitive, antisymmetric, and total binary relationship, $\succ_v$) over the candidates $C$, and gives 1 point to all candidates ordered in the top $c_v$ positions in $\pi_v$ and gives 0 point to all the other candidates, where $c_v$ is a positive integer associated to the voter $v$. The winner is a candidate who gets the highest total score. In addition, SP-AV has two important properties: admissible and sincere. The admissible property means that the first ordered candidate must be approved and the last ordered candidate must be disapproved, and the sincere property means that if a candidate is approved then all candidates ordered above him/she are also approved. Clearly, both properties are satisfied when $0 < c_v < |C|$. As showed in [3], compared to AV, SP-AV resists many types of attacks including constructive control by adding/deleting votes and constructive control by adding/deleting candidates, even when $r$ is a very small constant [23]. A slight variant of SP-AV election is the $r$-sincere-strategy preference-based approval ($r$-SP-AV) election, where each voter approves exactly his/her top $r$ candidates, that is, $c_v = r$ for all voters $v$.

Fixed parameterized complexity, introduced by Downey and Fellows [11], has been a powerful tool to deal with hard problems. Due to the fixed parameterized complexity, parameterized problems can be classified as follows.

$$\mathcal{FPT} \subseteq \mathcal{W}[1] \subseteq \mathcal{W}[2] \subseteq \ldots \subseteq \mathcal{W}[\text{SAT}] \subseteq \mathcal{W}[P] \subseteq \mathcal{XP}$$

where $\mathcal{FPT}$ includes all parameterized problems which admit $O(f(k) \cdot |I|^{O(1)})$-time algorithms, where $I$ is the input instance, $k$ is the parameter, and $f(k)$ is a computable function depending only on $k$. Many $\mathcal{FPT}$ problems and $\mathcal{FPT}$ algorithms have been found in the literature. We refer readers to [26, 19, 4] for more information on parameterized complexity.

In this paper, we focus on the complexity of strategic behaviors of $r$-SP-AV in $k$-peaked elections for different values of $k$. In this paper, we consider only constructive controls. Our main motivation is to fill the complexity gap between the single-peaked elections and the general case, that is, to establish a complexity dichotomy of control problems for $r$-SP-AV in $k$-peaked elections. In particular, we show that, control by adding votes for $r$-SP-AV with $r$ being a constant is polynomial-time solvable in 2-peaked elections, but $\mathcal{NP}$-hard in $k$-peaked elections for $k \geq 3$. Meanwhile, if $r$ is not a constant, then the problem of control by adding votes for $r$-SP-AV in 2-peaked elections becomes $\mathcal{NP}$-hard. Moreover, the deleting votes case turns out to be $\mathcal{NP}$-hard for $k$-peaked elections with $k \geq 2$, even for $r$ being a constant. Recall that all these control problems are polynomial-time solvable in single-peaked elections [16]. In addition, we prove that 1-SP-AV control by deleting candidates, in $k$-peaked elections for $k \geq 3$, is $\mathcal{NP}$-hard with respect to the number of deleted candidates. Finally, we study $r$-SP-AV control problems by adding/deleting votes in general from the viewpoint of parameterized complexity and prove that both problems are $\mathcal{FPT}$ with respect to the number of added and deleted votes, respectively. Our main results are showed in Table 1. Recall that $r$-SP-AV is a special case of SP-AV, all our $\mathcal{NP}$-hardness results apply to SP-AV.

---

1 In the original papers of [3, 13], the authors proved the $\mathcal{NP}$-hardness for SP-AV control by adding/deleting votes. The proofs can be easily modified to adopt to the case of $r$-SP-AV by changing from the exact $3$-cover problem to exact $r$-cover problem or exact $(r-1)$-cover problem (depends on whether it is control by adding candidates or deleting candidates) and changing each of the registered votes with many votes each of which approves exactly $r$ candidates such that each candidate gets the same score as in the proof in [3]. These can be done by first adding some dummy candidates if $r$ is not a factor of the number of candidates so that $r$ is a factor of the number of candidates, and then, dividing the candidates into many groups each of which contains $r$ candidates and are approved simultaneously.

2 In the literature [15], the authors studied a “nearly single-peaked election” called Swoon-SP election which is a special case of our 2-peaked election. These results marked by $\Delta$ are implicitly proved in the literature [15] when the authors studied the 1-swoop model. More details between these two models will be discussed in the related work section.
Then, a vote casted by \(v\) gives 1 point to all candidates ordered in \(\succ\) and gives 0 point to all other candidates. For convenience, we also use \(\succ\) to denote the partial vote \(\pi_v\) of a multiset \(\pi\), if, for every object \(s\) that occurs \(n\) times in \(A\), \(A\) contains at least \(n\) copies of \(s\). We use \(B \subseteq A\) to denote that \(B\) is a submultiset of \(A\).

\[\text{Example.}\] Consider multisets \(A = \{1, 1, 1, 2, 3, 3, 4\}\) and \(B = \{1, 2, 3\}\) then \(|A| = 7, |B| = 3, A \uplus B = \{1, 1, 1, 1, 2, 3, 3, 3, 4\}, A \ominus B = \{1, 1, 3, 4\}\) and \(B \subseteq A\).

\subsection*{r-sincere-strategy preference-based approval voting.} An \(r\)-sincere-strategy preference-based approval election (\(r\)-SP-AV for short) can be specified by a set \(C\) of candidates, a set \(V\) of voters where every \(v \in V\) casts a vote \(\pi_v\) which is defined as a linear ordering over the candidates \(C\). Each voter \(v\) gives 1 point to all candidates ordered in the top \(r\) positions in \(\pi_v\) and gives 0 point to all other candidates. For convenience, we also use \(\succ\) to denote \(\pi_v\). For a vote \(\succ_v\) and a candidate \(c\), the position of \(c\) in \(\succ_v\) is defined as \(|\{c' \in C | c' \succ_v c\}| + 1\). The multiset of votes casted by \(V\) is denoted by \(\Pi_V\). A winner is a candidate who gets the highest total score. If there is only one winner, we call it a unique winner.

\[\text{Example.}\] Consider a 2-SP-AV election specified by the candidates \(C = \{a, b, c\}\), the voters \(V = \{u, v, w\}\) where \(u\) casts the vote \(\pi_u = a \succ_v b \succ_v c, v\) casts the vote \(\pi_v = a \succ_v c \succ_v b\), and \(w\) casts the vote \(\pi_w = c \succ_v a \succ_v b\). Then, \(a\) would be the unique winner since \(a\) has the highest total score 3.

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

\subsection*{Single-peaked/k-peaked elections.} An election \((C, \Pi_V)\) is single-peaked if there is a linear ordering \(\mathcal{L}\) of \(C\) such that for every \(\succ_v\) in \(\Pi_V\) and every three candidates \(a, b, c \in C\) 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 \(\mathcal{L}\). Here, \(\mathcal{L}\) is called a harmonious ordering of \((C, \Pi_V)\), and the candidate ordered in the first position of \(\succ_v\) is called the peak of \(\succ_v\) with respect to \(\mathcal{L}\).

\[\text{Example.}\] Consider the election with candidates \(C = \{a, b, c, d, e\}\) and three votes \(\pi_u = b \succ_u d \succ_u e \succ_u c \succ_u a, \pi_v = d \succ_v b \succ_v c \succ_v a \succ_v e\) and \(\pi_w = a \succ_w c \succ_w b \succ_w d \succ_w e\). There is a harmonious ordering \(\mathcal{L}\) of \(C\) with \(a \mathcal{L} c \mathcal{L} b \mathcal{L} d \mathcal{L} e\). Thus, the election is a single-peaked election. See Fig 1 for a visual representation.
Fig. 1. An illustration of the single-peaked election with three votes \( \pi_u = b \succ u \ d \succ u \ c \succ u \ a \), \( \pi_v = d \succ v \ b \succ v \ c \succ v \ a \ v \ c \), and \( \pi_w = a \succ w \ c \succ w \ b \succ w \ d \succ w \ e \). The votes \( \pi_u \), \( \pi_v \), and \( \pi_w \) are illustrated by the dark line, the gray line, and the dotted line, respectively.

Fig. 2. A 2-peaked vote \( \pi_v = (c_3, c_4, c_7, c_6, c_9, c_5, c_2, c_{10}, c_1) \) with respect to an ordering \( L = (c_1, c_2, \ldots, c_{10}) \). It is clear that \( L \) can be partitioned into \( L_1 = (c_1, c_2, c_3, c_4, c_5) \) and \( L_2 = (c_6, c_7, c_8, c_9, c_{10}) \) such that \( \pi_v(C(L_1)) \) and \( \pi_v(C(L_2)) \) are single-peaked with respect to \( L_1 \) and \( L_2 \), respectively.

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

An election is \( k \)-peaked if there is an ordering \( L \) of \( C \) such that every vote in the election is \( k \)-peaked with respect to \( L \). Then \( L \) is called a \( k \)-harmonious ordering. Thus, 1-peaked elections are exactly the single-peaked elections and \( |C|/2 \)-peaked elections become the general case.

**Problem definitions.** The problems studied in this paper are defined as follows. Hereby, \( p \) denotes the distinguished candidate.

**r-SP-AV Control by Adding Votes in k-Peaked Elections (r-AV-k)**

*Input:* An \( r \)-SP-AV election \( (C \cup \{p\}, \Pi_Y) \) with a set \( \Pi_T \) of unregistered votes, where both \( \Pi_Y \) and \( \Pi_T \) are \( k \)-peaked with respect to a given \( k \)-harmonious ordering \( L \), and an integer \( 0 \leq R \leq |\Pi_T| \).

*Question:* Are there at most \( R \) votes \( \Pi_T \) in \( \Pi_Y \) such that \( p \) is the unique winner in the election \( (C \cup \{p\}, \Pi_Y \cup \Pi_T) \)?

**r-SP-AV Control by Deleting Votes in k-Peaked Elections (r-DV-k)**

*Input:* A \( k \)-peaked \( r \)-SP-AV election \( (C \cup \{p\}, \Pi_Y) \) associated with a \( k \)-harmonious ordering \( L \), and an integer \( 0 \leq R \leq |\Pi_Y| \).

*Question:* Are there at most \( R \) votes \( \Pi_T \) in \( \Pi_Y \) such that \( p \) is the unique winner in the election \( (C \cup \{p\}, \Pi_Y \cup \Pi_T) \)?

Further, we denote the problem of \( r \)-SP-AV control by adding and deleting votes in general by \( r \)-AV and \( r \)-DV, respectively.

**r-SP-AV Control by Adding Candidates in k-Peaked Elections (r-AC-k)**

*Input:* A \( k \)-peaked \( r \)-SP-AV election \( (C \cup D \cup \{p\}, \Pi_Y) \) associated with a \( k \)-harmonious ordering \( L \), and an integer \( 0 \leq R \leq |D| \).
Question: Are there at most $R$ candidates $D \subseteq D$ such that $p$ is the unique winner in $(C \cup D \cup \{p\}, \Pi_V(C \cup D \cup \{p\}))$?

$r$-SP-AV Control by Deleting Candidates in $k$-Peaked Elections ($r$-DC-$k$)

Input: A $k$-peaked $r$-SP-AV election $(C \cup \{p\}, \Pi_V)$ associated with a $k$-harmonious ordering $L$, and an integer $0 \leq R \leq |C|$.

Question: Are there at most $R$ candidates $C \subseteq C$ such that $p$ is the unique winner in $((C \cup \{p\}) \setminus C, \Pi_V((C \cup \{p\}) \setminus C))$?

1.2 Related Work

Complexity of AV control and SP-AV control problems in general have been studied in literature. In [20], the authors proved that AV control by adding/dedeting votes in general case are both $\mathcal{NP}$-hard (we also refer to [20] for complexity results of further AV control problems). A slight modification of the proofs for AV control by adding/deleting votes can be adopted to proving the $\mathcal{NP}$-hardness of SP-AV control by adding/deleting votes. This has been done in [3]. Furthermore, people studied these problems from the viewpoint of parameterized complexity and proved problems of AV and SP-AV Control by adding/deleting votes are $\mathcal{W}[1]$-hard, with respect to the number of added/deleted votes as the parameter [24]. In [23], Lin proved that 4-approval control by adding votes and 3-approval control by deleting votes in general case are both $\mathcal{NP}$-hard, while 3-approval control by adding votes and 2-approval control by deleting votes in general case are polynomial-time solvable. Note that these results also apply to SP-AV in general case, since preferences of the candidates are irrelevant to the problems of control by adding and deleting votes in general. As for the control by modification of candidates, AV turned out to be immune to control by adding candidates $^1$ and polynomial-time solvable for control by deleting case [20]. However, both problems of control by adding/deleting candidates are $\mathcal{NP}$-hard when restricted to SP-AV [3], even when degenerated to 1-SP-AV [2].

Single-peakedness was introduced by Black [6] in 1948. It has been shown that one can test whether a given election is a single-peaked election in polynomial time [14]. Moreover, a harmonious ordering can be found in polynomial time if the given election is a single-peaked election [14]. Recently, the complexity of many voting problems under single-peaked elections has been attracting much attention [8, 16, 5, 27]. In [16], people proved that the problems of AV and SP-AV control by adding/deleting votes, and the problems of 1-SP-AV control by adding/deleting candidates are polynomial-time solvable in single-peaked elections.

P. Faliszewski et al. [15] studied a nearly single-peaked election which is called Swoon-SP elections. An election is Swoon-SP if there is an ordering of candidates such that removing the first ordered candidate in every vote $v$ resulting in $v$ being single-peaked with respect to the ordering without the deleted candidate. Clearly, the Swoon-SP election is a special case of 2-peaked election. They proved that the problems of 1-SP-AV control by adding/deleting candidates are $\mathcal{NP}$-hard when restricted to the Swoon-SP elections. For further variations of single-peaked elections we refer to [15, 12, 10, 9].

2 2-Peaked Elections

In this section, we study $r$-SP-AV control problems in 2-peaked elections. The following three theorems summarize our findings.

Theorem 1. $r$-AV-2 is polynomial-time solvable for any constant $r$.

Recall that $r$-AV is $\mathcal{NP}$-hard for any constant $r \geq 4$ but polynomial-time solvable when restricted to single-peaked elections [16]. The above theorem 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 the number of approved candidates is part of the input, then $r$-AV becomes $\mathcal{NP}$-hard in 2-peaked elections, in contrast to the polynomial-time solvability in the single-peaked case [16].

$^1$ A voting system is immune to a constructive strategic behavior if one cannot make the candidate who is not a winner become a final winner by imposing the strategic behavior into the election.
Theorem 2. \( r \)-AV-2 is \( \mathcal{NP} \)-hard, when \( r \) is part of the input.

The problem of control by deleting votes for \( r \)-SP-AV has been proved polynomial-time solvable for even non-constant \( r \) [16]. The following theorem shows that by increasing the number of peaks only by one, this problem becomes \( \mathcal{NP} \)-hard for even \( r \) is a constant.

Theorem 3. \( r \)-DV-2 is \( \mathcal{NP} \)-hard for any constant \( r \geq 3 \).

2.1 Proof of Theorem 1

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

Let \( (\mathcal{C} \cup \{p\}, \Pi_Y, \Pi_T, \mathcal{L}, R) \) be an instance of \( r \)-AV-2. For \( c \in \mathcal{C} \), let \( \mathcal{C}(1) \) be the candidate lying immediately before \( c \) in \( \mathcal{L} \) and \( \mathcal{C}(i) \) be the candidate lying immediately before \( \mathcal{C}(i-1) \) in \( \mathcal{L} \). Similarly, we use \( \mathcal{C}(1) \) and \( \mathcal{C}(i) \) to denote the candidates lying immediately after \( c \) and \( \mathcal{C}(i-1) \), respectively. For example, if \( \mathcal{L} = (a, b, c, d, e, f, g, h) \), then \( \mathcal{C}(1) = e \), \( \mathcal{C}(4) = h \), \( \mathcal{C}(1) = c \) and \( \mathcal{C}(3) = a \). Moreover, we define \( \mathcal{C}(i,j) = \{ \mathcal{C}(k) \mid i \leq k \leq j \} \) and \( \mathcal{C}(i,j) = \{ \mathcal{C}(k) \mid i \leq k \leq j \} \).

For a vote \( \pi \), let \( I_1 \) denote the set of candidates who get 1 point from \( \pi \) and 0 point from \( \pi \). For a candidate \( c \), let \( SC_V(c) \) be the total score of \( c \) from \( \Pi_Y \), that is, \( SC_V(c) = \{ \{ \pi \in \Pi_Y \mid c \in 1(\pi) \} \} \).

Given an ordering \( A = (a_1, a_2, \ldots, a_n) \), a discrete interval \( I \) over \( A \) is a consecutive sub-ordering \( (a_i, a_{i+1}, \ldots, a_{i+t}) \) 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 \( S(I) \) denote the set of elements appearing in \( I \) and set \( |I| := |S(I)| \). For example, for a discrete interval \( I = (3, 6) \) over the ordering \( A = (2, 5, 3, 10, 4, 6, 0) \), \( S(I) = \{3, 4, 6, 10\} \). A \( k \)-discrete interval over an ordering \( A \) 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 \( I \), let \( S(I) = \bigcup_{I \in I} S(I) \).

Observation 4 For each \( k \)-peaked election \((\mathcal{C}, \Pi_Y)\) associated with a \( k \)-harmonious ordering \( \mathcal{L} \) over \( \mathcal{C} \), and each vote \( \pi_v \in \Pi_Y \), there is a \( k' \)-discrete interval \( I \) over \( \mathcal{L} \) such that \( 0 < k' \leq k \) and \( 1(\pi) = S(I) \).

Due to Observation 4, for each vote \( \pi_v \), in a 2-peaked election associated with \( \mathcal{L} \) as a 2-harmonious ordering, \( 1(\pi) \) can be represented by a 2-discrete interval or a 1-discrete interval over \( \mathcal{L} \). See Fig. 3 for an example.

![Fig. 3. 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 \( \pi_u = (c_7, c_6, c_5, c_8, c_9, c_{10}, c_1, c_4, c_3, c_2) \) in a 2-peaked election. Each vote gives 1 point to its top four ordered candidates and 0 point to other candidates. It shows that \( 1(\pi) \) can be represented by a 2-discrete interval \( \{I_v = (c_3, c_4), I_u = (c_6, c_7)\} \) and \( 1(u) \) can be represented by a 1-discrete interval \( \{I_u = (c_5, c_6, c_7, c_8)\} \). Here we use dark lines to connect the elements appearing in the same discrete interval. However, they do not belong to the discrete intervals.](image)

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 5 Every true-instance of \( r \)-AV has a solution where each added vote approves the distinguished candidate \( p \).
Due to Observation 5, we can safely assume that for every \( \pi_v \in \Pi_T \), \( p \in \Pi_T \), \( v \in (v) \). By Observation 4, for each vote \( \pi_v \in \Pi_T \), \( 1(v) \) can be represented by a 2-discrete interval \( \mathcal{I}_v = \{ I_v^p, I_v^p \} \) or a 1-discrete interval \( \mathcal{I}_v = \{ I_v^p \} \) with \( p \in \mathcal{S}(I_v^p) \). Let \( S \) be the multiset of all votes \( \pi_v \in \Pi_T \) where \( 1(v) \) is represented by a 1-discrete interval over \( \mathcal{I} \). 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 \( \bar{\pi}(3), \bar{\pi}(2), \bar{\pi}(1), p \);
2. votes approving \( \bar{\pi}(2), \bar{\pi}(1), p, \bar{\pi}(1) \);
3. votes approving \( \bar{\pi}(1), p, \bar{\pi}(1), \bar{\pi}(2) \); and
4. votes approving \( p, \bar{\pi}(1), \bar{\pi}(2), \bar{\pi}(3) \).

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 \( \Pi_T \) is represented by a 2-discrete interval. Let \( \Pi_T = (\pi_v_1, \pi_v_2, ..., \pi_v_{|\Pi_T|}) \) be an ordering of \( \Pi_T \) such that \( r(I_v^p) = r(I_v^p) \) or \( r(I_v^p) \) \( r(I_v^p) \) for all \( 1 \leq i < j \leq |\Pi_T| \).

Our dynamic programming algorithm is based on a dynamic programming associated with a binary dynamic table \( DT(i, j, k, s_1, s_2, s_3, s_4, s_5, s_6, s_{i,1}, s_{i,2}, s_{i,3}) \), where we set \( DT(i, j, k, s_1, s_2, s_3, s_4, s_5, s_6, s_{i,1}, s_{i,2}, s_{i,3}) = 1 \) if there is a submultiset \( \Pi_T \subseteq \{ \pi_v_1, \pi_v_2, ..., \pi_v_i \} \) satisfying

1. \( |\Pi_T| = j; \)
2. \( \pi_v_i \in \Pi_T; \)
3. \( \text{max}\{SC_{\Pi_T}(c) \mid c \in C\} = k; \)
4. \( SC_{\Pi_T}(c_i) = s_t \) for all \( 1 \leq t \leq 6 \), where \( \bar{\pi}(1), c_2 = \bar{\pi}(2), c_1 = \bar{\pi}(3), c_4 = \bar{\pi}(1), c_5 = \bar{\pi}(2), c_6 = \bar{\pi}(3) \); and
5. \( SC_{\Pi_T}(c_{i,t}) = s_{i,t} \) for all \( t \in \{1, 2, 3\} \), where \( c_{i,1} = r(I_v^p), c_{i,2} = \bar{\pi}(1), c_{i,3} = \bar{\pi}(2) \).

(See Fig. 4 for an illustration of (4) and (5)).

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

We use the following iterative recurrence to update the table.

**Case 1.** \( DT(i, j, k, s_1, s_2, ..., s_6, s_{i,1}, s_{i,2}, s_{i,3}) = 1 \), if at least one of the following cases applies:

**Case 1.** \( DT(i, j, k, s_1, s_2, ..., s_6, s_{i,1}, s_{i,2}, s_{i,3}) = 1 \) such that:

1. \( j - 1 \leq i_1 \leq i - 1; \)
2. \( s_1 = s_1' + SC_{\{s_1\}}(c_1) \) for all \( 1 \leq t \leq 6; \)
3. \( s_{i,t} = s_{i,t'} + SC_{\{s_i\}}(c_i) \) for all \( 1 \leq t \leq i_1 \leq 3 \) with \( c_i = c_i,t \); and
4. \( s_{i,t} = SC_{\Pi_T}(c_i) \) for all \( c_i \in \{ r(I_v^p), r(I_v^p),(1), r(I_v^p),(2) \} \). 

**Case 2.** \( \exists s \in \{ s_1, s_2, ..., s_6, s_{i,1}, s_{i,2}, s_{i,3} \} \) with \( s = k \) and \( \exists DT(i, j, k, s, s_1', s_2', ..., s_6', s_{i,1}', s_{i,2}', s_{i,3}') \) such that:

1. \( j - 1 \leq i_1 \leq i - 1; \)
2. \( s_1 = s_1' + SC_{\{s_1\}}(c_1) \) for all \( 1 \leq t \leq 6; \)
3. \( s_{i,t} = s_{i,t'} + SC_{\{s_i\}}(c_i) \) for all \( 1 \leq t \leq i_1 \leq 3 \) with \( c_i = c_i,t \); and
4. \( s_{i,t} = SC_{\Pi_T}(c_i) \) for all \( c_i \in \{ r(I_v^p), r(I_v^p),(1), r(I_v^p),(2) \} \). 

**Fig. 4.** Illustration of (4) and (5) in the definition of \( DT \).
The algorithm is easy to be generalized for \( r \geq 4 \) being a constant by using a bigger but still polynomial-sized dynamic table.

2.2 Proof of Theorem 2

In this section we prove Theorem 2 by reducing a variant of INDEPENDENT SET, which has been proved \( \mathcal{NP} \)-hard [22] to the \( r \)-AV-2 problem. We begin with some definitions.

An empty ordering denoted by ( ) is a linear order containing no element. For a linear order \( A = (a_1, a_2, ..., a_n) \) over the set \( \{a_1, a_2, ..., a_n\} \), let \( A[a_i, a_j] \) (resp. \( A(a_i, a_j) \), \( A(a_i, a_j) \)) with \( i \leq j \) be the sub-order \( (a_i, a_{i+1}, ..., a_j) \) (resp. \( (a_i, a_{i+1}, ..., a_j) \)) if \( i < j \) and ( ) if \( i = j \), \( (a_i, a_{i+1}, ..., a_{j-1}) \) if \( i < j \) and ( ) if \( j \geq i \geq j - 1 \), and let \( A[a_j, a_i] \) (resp. \( A(a_j, a_i) \), \( A(a_j, a_i) \)) be the order of \( A[a_i, a_j] \) (resp. \( A(a_i, a_j) \), \( A(a_j, a_i) \)). For two linear orders \( A = (a_1, a_2, ..., a_n) \) over set \( A' \) and \( B = (b_1, b_2, ..., b_m) \) over set \( B' \) with \( A' \cap B' = \emptyset \), we denote by \( (A, B) \) the linear order \( (a_1, a_2, ..., a_n, b_1, b_2, ..., b_m) \). For an positive integer \( n \), let \( [n] = \{1, 2, ..., n\} \). In the following, let \( \mathcal{N} \) be the ordering \((1, 2, ..., n)\) of all positive integers.

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 \( \mathcal{N} \) and \( |T_i| \leq 3 \) for all \( T_i \in \mathcal{T} \).

Question: Is there a set \( S \subseteq \bigcup_{T_i \in \mathcal{T}} T_i \) 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 \( \mathcal{E} = (\mathcal{T} = \{T_1, T_2, ..., T_n\}) \) of VIS, we construct an instance \( \mathcal{E}' = ((\mathcal{C} \cup \{p\}), \Pi_\mathcal{V}, \Pi_{\mathcal{T}}, \mathcal{L}, \mathcal{R}) \) for \( r \)-AV-2 as follows.

Let \( \mathcal{I} = \bigcup_{T_i \in \mathcal{T}} T_i \). For each discrete interval \( I \in \mathcal{I} \), let \( l(I) \) be its left endpoints and \( r(I) \) be its right endpoints. Let \( \Gamma \) be the set of all elements appearing in some discrete interval of \( \mathcal{I} \), i.e., \( \Gamma = \{S(I) \mid I \in \mathcal{I}\} \). Let \( \Gamma = (x_1, x_2, ..., x_{|\Gamma|}) \) be an ordering of \( \Gamma \) 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 \( d_1, d_2, ..., d_n, ..., d_{2n-1} \); (3) \( E \) contains exactly \( (n + 3) \cdot (|C| + |D| - 1) \) dummy candidates \( x'_1, x'_2, ..., x'_{|C|+1}, d'_1, d'_2, ..., d'_{|C|+(|D|-1)} \) which will never be winners. The distinguished candidate is \( d_n \), that is, \( p = d_n \), \( r = n + 4 \).

2-Harmonious Ordering: \( \mathcal{L} = (\Gamma, D, E) \) where \( D = (d_1, d_2, ..., d_{2n-1}) \) and \( E = (x'_1, x'_2, ..., x'_{|C|+1}, d'_1, d'_2, ..., d'_{|C|+(|D|-1)}) \).

Registered Votes \( \Pi_{\mathcal{H}} \): We create the following registered votes: (1) for each \( x_i \in C \), create \( n - 2 \) votes defined as \( \{x_i, \mathcal{L}[(x'_j, (x'_j+1)-n-2), (x'_j+1)-n-2), \mathcal{L}(x_i, x_1), \mathcal{L}(x_i, x'_j), \mathcal{L}(x_i, x'_j+1)-n-2), \mathcal{L}(x'_j, (x'_j+1)-n-2), \mathcal{L}(x'_j+1)-n-2), \mathcal{L}(x'_j+1)-n-2), \mathcal{L}(x'_j+1)-n-2), \mathcal{L}(x'_j+1)-n-2), \mathcal{L}(x'_j+1)-n-2) \). (2) for each \( d_i \in D \) where \( i \in [n-1] \), create \( n - (i + 1) \) votes defined as \( \{d_i, \mathcal{L}[d'_i, (d'_i+1)-n-2), d'_i, \mathcal{L}(d_i, x_1), \mathcal{L}(d_i, d'_i), \mathcal{L}(d'_i, (d'_i+1)-n-2), \mathcal{L}(d'_i+1)-n-2), \mathcal{L}(d'_i+1)-n-2), \mathcal{L}(d'_i+1)-n-2), \mathcal{L}(d'_i+1)-n-2), \mathcal{L}(d'_i+1)-n-2), \mathcal{L}(d'_i+1)-n-2) \). (3) for each \( d_i \in D \) where \( i \in \{n+1, n+2, ..., 2n-1\} \), create \( i - (n + 1) \) votes which is defined as \( \{d_i, \mathcal{L}[d'_i, (d'_i+1)-n-5), d'_i, \mathcal{L}(d'_i, (d'_i+1)-n-5), \mathcal{L}(d'_i+1)-n-5), \mathcal{L}(d'_i+1)-n-5), \mathcal{L}(d'_i+1)-n-5), \mathcal{L}(d'_i+1)-n-5), \mathcal{L}(d'_i+1)-n-5) \} \).

Unregistered Votes \( \Pi_{\mathcal{U}} \): For each \( I_{ij} \in T_i \in \mathcal{T} \), create a corresponding unregistered vote which is defined as \( \{l(I_{ij}), \mathcal{L}[l(I_{ij}), \mathcal{L}[d_i, d'_i, \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 unregistered votes are 2-peakled with respect to \( \mathcal{L} \).

Number of Added Votes: \( R = n \).

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

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 \( i \in \{n+1, n+2, ..., 2n-1\} \), \( SC_{\mathcal{V}}(c) \leq n - 2 \) for all \( c \in E \) and \( SC_{\mathcal{V}}(d_i) = 0 \).

\( \Rightarrow \): Suppose that \( \mathcal{E} \) is a true-instance and let \( S \) be a solution for \( \mathcal{E} \). Let \( \mathcal{S} = (I_1, I_2, ..., I_n) \) be an ordering 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 $\pi_v$ and $\pi_u$ which corresponds 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 $d_n$. 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 $SC_V(d_j) = n - j - 1$ 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 $SC_V(d_j) = j - n - 1$ and $|S \cap T_i| = 1$ for all $i \in \{j - n + 1, j - n + 2, \ldots, n\}$, we know that the final score of $d_j$ is less than $n$. The final score of any $c \in E$ is clearly less than $n - 2$ since no unregistered vote approves $c$. Summarize the above analysis, we conclude that the distinguished candidate $d_n$ become the unique winner after adding the selected votes to the election.

$\Leftarrow$: Suppose that $E'$ is a true-instance and $S'$ is a multiset of sets chosen from $II_T$ which makes $d_n$ the unique winner in the election $(C \cup D \cup E, II_V \cup S')$. It is easy to verify that $|S'| = n$; 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 $II_T$ where $P_i$ contains all votes corresponding to the intervals of $T_i \in T$. Clearly, $P_i \cup V$ is a set. We claim here that $|S' \cap P_i| = 1$. Suppose this is not true, then there must be a $P_i$ 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,

$$SC_{\cup} S'(d_i) + SC_{\cup} S'(d_{i+n-1}) = SC_V(d_i) + |S_1| + |S_2| + SC_V(d_{i+n-1}) + |S_1| + |S_3| = n - i - 1 + |S_1| + |S_2| + i - 2 + |S_1| + |S_3| = 2n - 3 + |S_1| \geq 2n - 1$$

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. Thus, the claim is true.

It is now easy to see that $S = \{I(v) \mid \pi_v \in S'\}$ is a solution for $E$, where $I(v)$ is the discrete interval corresponding to $\pi_v$.

2.3 Proof of Theorem 3

We first prove that 3-DV-2 is $\mathcal{NP}$-hard by a reduction from Vertex Cover on graphs with degree bounded by 3 which is $\mathcal{NP}$-hard [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 $v \in V$, $N_G(v)$ is used to denote its neighbors in $G$, that is, $N_G(v) = \{w \mid (w, v) \in E\}$. 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 $k$.

Question: Does $G$ have a vertex cover of size at most $k$?

In the following, we first introduce a property for bounded degree-3 graphs. Then, we will use this property to prove the $\mathcal{NP}$-hardness of 3-DV-2.

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. For an interval $I := [a, b]$, we use $r(I)$ and $l(I)$ to denote the right-endpoint and the left-endpoint of $I$, that is, $r(I) = b$ and $l(I) = a$. A $t$-interval is a set of no more than $t$ intervals over the real line. A graph $G = (V, E)$ is a $t$-interval graph if there is a set $T_G$ of $t$-intervals and a bijection $f : V \rightarrow T_G$ such that for every $u, w \in V$, $(u, w) \in E$ if and only if $f(u)$ and $f(w)$ intersect. Here, $(T_G, f)$ is called a $t$-interval representation of $G$. We also use $T_G$, for short, to denote a $t$-interval representation of the graph $G$ when it is clear from the context. For
simplicity, we use $I_u = \{I^1_u, I^2_u, ..., I^t_u\}$ with $t' \leq t$ to denote $f(u)$. For two real numbers $a$ and $b$ with $a \leq b$, we define $(a, b) = \{x \in \mathbb{R} \mid a < x < b\}$.

In the paper [18], the authors proved that any graph with bounded degree-$\delta$ has a $[\frac{1}{2}(\delta + 1)]$-interval representation. In below, we first introduce another more precise property, which will be very useful for our proof, for graphs with bounded degree-3. This property says that all graphs of bounded degree-3 has a 2-interval representation with some special conditions.

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

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

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

where $I(u') \in \{I^1_u, I^2_u\}$. Moreover, such a 2-interval representation can be found in polynomial time.

In the following, we show the reduction. Let $\mathcal{E} = (G, k)$ be an instance of VC3. Let $\mathcal{I}(G)$ be a 2-interval representation of $G$ satisfying all conditions in Lemma 1. For every $I_u = \{I^1_u, I^2_u\}$, let $D(u)$ be the endpoints of $I^1_u$ and $I^2_u$ (recall that by Lemma 1, $|D(u)| = 3$ for all $u \in V(G)$), and let $\Gamma = \cup_{u \in V(G)} D(u)$. Let $\mathcal{I} = (\{x_1, x_2, ..., x_{|\Gamma|}\}$ be the ordering of $\Gamma$ with $x_i < x_{i+1}$ for all $i \in [|\Gamma| - 1]$. We construct an instance $\mathcal{E}' = (\{\cup \{p, r\}, \Pi(u), R, L\}$ 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 Ordering:** $\mathcal{L} = (\mathcal{I}, p, c_1, c_2, c_3, c_4)$.

**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 a $I_u$ in $\mathcal{I}(G)$ for $u \in V(G)$. More specifically, for every $I_u$, let $(x_i, x_j, x_k)$ be the ordering of $D(u)$ with $x_i < x_j < x_k$, then we create a vote $\pi_u = (x_i, x_j, x_k, L(x_i, x_j), L(x_i, x_k), L(x_j, x_k), L(x_k, c_4))$. Thus, $\pi_u$ approves $D(u)$. Due to Lemma 1, either $x_i$ or $x_k$ lies consecutively with $x_j$ in $\mathcal{L}$, that is, one of $x_i = x_j^1(1)$ and $x_k = x_j^2(1)$ 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: $(p, c_1, c_2, c_3, c_4, L(p, x_1))$ and $(p, c_3, c_4, c_1, c_2, L(p, x_1))$. It is clear that these two votes are 2-peaked with respect to $\mathcal{L}$.

**Number of Deleted Votes:** $R = k$.

The following observation is clearly true.

**Observation 6** Every true-instance of 3-DV has a solution where all deleted votes do not approve $p$.

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 $k$ 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. Due to Observation 6, there must be a solution $S'$ of size at most $k$ such that all votes in $S'$ do not approve $p$. Therefore, $p$ has two points in the final election, and every other candidate can have at most one point after deleting all votes of $S'$, which implies that no two votes of the first type can approve a common candidate in $\mathcal{C}$ after deleting all votes from $S'$, further implying that the vertices corresponding to $S'$ form a vertex cover of size at most $k$ for $G$.

In order to prove that r-DV-2 is $NP$-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 $X_i = \{x^1_i, x^2_i, ..., x^t_i\}$ with the order $(x^1_i, x^2_i, ..., x^t_i)$ between $x_j \in \Gamma$ and $x_{j+1} \in \Gamma$ in the 2-harmonious ordering $\mathcal{L}$ whenever there is a $u \in V(G)$ such that $[x_i, x_{i+1}] \in I_u$. 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 $I_u = \{[x_i, x_{i+1}], [x_j, x_j]\}$ (resp. $I_u = \{[x_i, x_j], [x_j, x_{j+1}]\}$), we create a vote defined as $(L(x_i, x_{i+1}), x_j, L(x_i, x_{i+1}), L(x_j, x_{j+1}), L(x_j, x_j), L(x_j, x_{j+1}), L(x_j, c_{2r-2}))$. As for the second type of votes, we have still two votes defined as $(L(p, c_{2r-2}), L(p, x_1))$ and $(p, L(c_5, c_6, ..., c_{2r-2}), L(c_5, c_6, ..., c_{2r-2}), L(p, x_1))$, respectively. Then, with the same argument, we can show that r-DV-2 is $NP$-hard for any constant $r \geq 4$. 

10
3 3-Peaked Elections

In Section 2, we proved that the problem of control by adding votes for r-SP-AV is polynomial-time solvable when restricted to 2-peaked elections for r being a constant. In this section, we show that the tractability of the problem does not hold when extended to 3-peaked elections. In addition, we study the problem of control by deleting candidates for 1-SP-AV and prove that the problem is \( W[1] \)-hard with respect to the number of deleted candidates.

**Theorem 7.** r-AV-3 is \( \mathcal{NP} \)-hard for any constant \( r \geq 4 \).

The 1-DC-3 has been proved \( \mathcal{NP} \)-hard [15]. The following theorem shows that this problem is \( W[1] \)-hard.

**Theorem 8.** 1-DC-3 is \( W[1] \)-hard with respect to the number of deleted candidates.

3.1 Proof of Theorem 7

We first prove the \( \mathcal{NP} \)-hardness of 4-AV-3 by a reduction from Independent Set on bounded degree-3 graphs which has been proved \( \mathcal{NP} \)-hard [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 \( S \).

Indefinite Set on Bounded Degree-3 graphs (IS3)

**Input:** A bounded degree-3 graph \( G = (V, E) \) and a positive integer \( k \).

**Question:** Does \( G \) have an independent set containing exactly \( k \) vertices?

For an instance \( E = (G, k) \) of IS3, let \( \mathcal{I}(G) \) be a 2-interval representation of \( G \) which satisfies all conditions in Lemma 1. Let \( D(u), \Gamma \) and \( \Gamma' \) be defined as in Subsect. 2.3. We construct an instance \( E' = ((\mathcal{C} \cup \{p\}, \mathcal{I}(\mathcal{V})), \Pi', L, R) \) of 4-AV-3 as follows.

**Candidates:** \( C = \Gamma \cup \{p, c_1, c_2, c_3\} \).

**3-Harmonious Ordering:** \( \mathcal{L} = (\Gamma, p, c_1, c_2, c_3) \).

**Registered Votes** \( \Pi' \): The role of registered votes is to guarantee that all candidates of \( \Gamma \) have the same score \( k - 2 \). To this end, we first create \( k - 2 \) votes defined as \( (\mathcal{L}[x_i, x_{i+1}], \mathcal{L}(x_i, x_1), \mathcal{L}(x_{i+3}, c_3)) \) for every \( i = 1, 5, \ldots, 4[|\Gamma|/4] - 3 \). Then, we create some other votes according to \( |\Gamma| \).

Case 1. \( |\Gamma| \equiv 0 \mod 4 \). We create no other vote.

Case 2. \( |\Gamma| \equiv 1 \mod 4 \). We create additional \( k - 2 \) votes defined as \( (\mathcal{L}[x_{\Gamma}], \mathcal{L}(c_1, c_3), \mathcal{L}(x_{\Gamma}, x_1), p) \).

Case 3. \( |\Gamma| \equiv 2 \mod 4 \). We create additional \( k - 2 \) votes defined as \( (\mathcal{L}[x_{\Gamma-1}, x_{\Gamma}], c_1, c_3, \mathcal{L}(x_{\Gamma-1}, x_1), p, c_3) \).

Case 4. \( |\Gamma| \equiv 3 \mod 4 \). We create additional \( k - 2 \) 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** \( \Pi' \): We create \( |V(G)| \) unregistered votes. For each \( u \in V(G) \), let \( (x_i, x_j, x_k) \) be the ordering of \( D(u) \) with \( x_i < x_j < x_k \). We create a vote \( \pi_u = (x_i, x_j, x_k, p, \mathcal{L}(x_1, x_i), \mathcal{L}(x_1, x_j), \mathcal{L}(x_1, x_k), \mathcal{L}(x_k, p), \mathcal{L}(p, c_3)) \). Due to Lemma 1, either \( x_i \) or \( x_k \) lies consecutively with \( x_j \) in \( \mathcal{L} \); thus, all these unregistered votes have 3 peaks \( x_\alpha, x_\beta \) 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 \( \mathcal{L} \).

**Number of Added Votes:** \( R = k \).

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

\( \Rightarrow: \) Suppose that \( E \) is a true-instance and \( S \) is an independent set of size \( k \). 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 \( k - 1 \) points. Since each added vote approves \( p, p \) has a final score of \( k \) points, which implies that \( p \) becomes the unique winner in the election including the new votes.

\( \Leftarrow: \) Suppose that \( E' \) is a true-instance and \( S' \) is a solution. Clearly, \( p \) has a final score of \( k \) points. Since \( p \) is the unique winner, for every \( c \in \mathcal{C} \setminus \{p, c_1, c_2, c_3\} \), 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 can apply to r-AV-3 for any constant \( r \geq 5 \) by a similar modification as discussed in Subsect. 2.2.
3.2 Proof of Theorem 8

In this section, we prove that 1-DC-3 is $\mathcal{W}[1]$-hard by a reduction from INDEPENDENT SET a $\mathcal{W}[1]$-hard problem [11] where we are given a graph $G$ and a positive integer $k$, and asked whether there is a subset of $V(G)$ such that no two vertices in this subset are adjacent.

For a linear order $A = (a_1, a_2, ..., a_n)$ over the set $A = \{a_1, a_2, ..., a_n\}$ and a subset $B \subseteq A$, denote by $A \setminus B$ the linear order of $A \setminus B$ obtained from $A$ by deleting all elements in $B$. For example, for $A = \{2, 7, 6, 4, 3, 9\}$ and $B = \{7, 4\}$, $A \setminus B = \{2, 6, 3, 9\}$.

For an instance $E = (G, k)$ of Independent Set we construct an instance $E'$ of 1-DC-3 as follows.

**Candidates:** $V \cup \{p, a, a_1, a_2, ..., a_k, b, b_1, b_2, ..., b_k\}$.

**3-Harmonious Ordering:** Let $F = (c_1, c_2, ..., c_n)$ be an (arbitrary) ordering of $V$. Then, the 3-harmonious ordering $L$ is given by $(b_k, b_{k-1}, ..., b_1, b, a, a_1, a_2, ..., a_k, c_1, c_2, ..., c_n)$.

**Votes:** There are seven types of votes. (1) $2m - 1$ votes defined as $(L[a, c_n], L[p, b_k])$; (2) $2m$ votes defined as $(L[p, c_n], L[b, b_k])$; (3) $2m + k - 1$ votes defined as $(L[b, b_k], L[p, c_n])$; (4) for each edge $(c_i, c_j) \in E(G)$ with $i < j$, create one vote defined as $(c_i, c_j, L[a, a_k], L[p, b_k], F \setminus \{c_i, c_j\})$; (5) for each vertex $c_i$, create one vote defined as $(c_i, L[p, a_k], L[b, b_k], F \setminus \{c_i\})$ and one vote defined as $(c_i, L[a, a_k], L[p, b_k], F \setminus \{c_i\})$; (6) $k + 1$ votes defined as $(L[a_1, c_1], L[a, b_k])$; (7) one vote defined as $(L[b_1, b_k], L[b, c_n])$; It is easy to verify that all constructed votes are 3-peaked with respect to $L$.

**Number of Added Candidates:** $R = k$.

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

In the following, we show the correctness of the other direction. Suppose that $E'$ is a true instance and $S'$ with $|S'| \leq k$ is a solution. We first observe that $b \not\in S'$. This observation is true, since otherwise, all candidates in $\{b_1, b_2, ..., b_k\}$ must be deleted, contradicting that $|S'| \leq k$. The same argue applies to the candidate $a$. It is easy to see that $S'$ does not contain any of $\{a_1, a_2, ..., a_k, b_1, b_2, ..., b_k\}$, since otherwise, $p$ cannot beat $b$. Thus, $S'$ contains only candidates from $V$. It is now easy to verify that any two $c_1, c_2 \in S'$ cannot adjacent to each other in the graph $G$, since otherwise, $a$ would get at least one extra point from the constructed votes of case 4 and then $p$ cannot be the unique winner. Thus, $S'$ forms an independent set. It is then easy to see that $|S'| = k$, since otherwise, $b$ would be the winner, which completes the proof.

4 Parameterized Complexity

In this section, we study the problem of control by adding/deleting votes for $r$-SP-AV in general case ($r$-AV/$r$-DV) from the viewpoint of parameterized complexity. Recall that both problems of control by adding/deleting votes for SP-AV are $\mathcal{W}[1]$-hard [24] with respect to the number of added/deleted votes as the parameter. The following theorems, however, show that these problems become $\mathcal{FPT}$ when restricted that every vote approves exactly $r$ candidates for $r$ being a constant.

**Theorem 9.** $r$-DV is $\mathcal{FPT}$ with respect to the number of deleted votes as the parameter, where $r$ is a constant.

**Proof.** To derive the $\mathcal{FPT}$ algorithm, we divide the candidates $C$ into two parts: $C_1 = \{c \in C \mid SC_V(c) \geq SC_V(p)\}$ and $C_2 = C \setminus C_1$. Meanwhile, we divide $\Pi_V$ into two parts: $\Pi_{V_1} = \{\pi_v \in \Pi_V \mid \exists c \in C_1 \text{ with } c \in 1(v)\}$ and $\Pi_{V_2} = \Pi_V \setminus \Pi_{V_1}$. The following observation is clearly true.

**Observation 10** For every true-instance of $r$-DV, there must be a solution $S$ such that $S \subseteq \Pi_{V_1}$.

Due to Observation 10, we can restrict our attention to $\Pi_{V_1}$. Since we can remove at most $R$ votes and each vote can approve at most $r$ candidates in $C_1$, $|C_1| \leq r \cdot R$ holds for every true-instance. Thus, we can safely assume that the given instance has at most $r \cdot R$ candidates in $C_1$.

We say two votes are equal or 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 $\Pi_{V_1}$. Since every solution includes at most $R$ votes from each type of votes, we have at most $R^{O((R))^r}$ cases to check where $f(R) = R^r$, implying an $\mathcal{FPT}$ algorithm for $r$-DV.

**Theorem 11.** $r$-AV is $\mathcal{FPT}$ with respect to the number of added votes as the parameter, where $r$ is a constant.
To prove the theorem, we reduce this problem to a generalized r-set packing problem which has been studied in [25]. Then, we derive an FPT algorithm for this generalized r-set packing problem.

In the r-set packing problem, we are given a set \( C \) of elements and a collection \( V \) of \( r \)-subsets of \( C \), and we want to know whether there is a size-\( R \) sub-collection \( T \) of \( V \) such that every element in \( C \) occurs in at most one \( r \)-subset in \( T \). Jia et al. [21] derived a smart FPT algorithm for r-set packing for any constant \( r \). Compared with r-set packing, the generalized r-set packing allows that every element \( c_i \) can occur in at most \( b_i \) many \( r \)-subsets in \( T \). Let \( \mathbb{N}^+ \) be the set of all positive integers.

Multi-r-set Packing, MrSP

Input: An universal set \( C \), a multiset \( V \) of \( r \)-subset of \( C \), a mapping \( f : C \to \mathbb{N}^+ \) with \( f(c) \leq c_V \) for all \( c \in C \) where \( c_V \) is the number of the occurrences of \( c \) in \( V \), and a positive integer \( R \).

Question: Is there a submultiset \( T \) with \( |T| = R \) of \( V \) such that every \( c \in C \) appears in at most \( f(c) \) elements of \( T \)?

Such a \( T \) is called a f-r-set packing of size \( R \).

In the following, we give an FPT algorithm for MrSP for \( r \) being a constant, based on the algorithm proposed by Weijia et al. [21].

We use the same notations in [21]. A f-r-set packing \( T \) is maximal if adding any set from \( V \) to \( T \) will make some \( c \in C \) occur in more than \( f(c) \) sets in \( T \). A partial set \( \sigma^* \) is a set in \( V \) with zero or more elements in \( \sigma^* \) replaced by the symbol \( * \). A set without \( * \) is also called a regular set. The set consisting of the non-* elements in a partial set \( \sigma^* \) is denoted as \( \text{reg}(\sigma^*) \). A partial set \( \sigma^* \) is consistent with a regular set \( \sigma \) if \( \text{reg}(\sigma^*) \subseteq \sigma \). For an f-r-set packing \( T \), let \( S(T) \) be the set of objects contained in some \( r \)-subset of \( C \) in \( T \). We call an ordering \( (c_{a_1}, c_{a_2}, \ldots, c_{a_i}) \), where each \( c_{a_i} \) is from \( C \) and \( a_i = a_j \) is allowed, is "valid" if for any \( c \in C \) there are at most \( f(c) \) copies of \( c \) appearing in the ordering. A multiset \( T' \) of partial sets is called valid if any \( c \in C \) occurs in at most \( f(c) \) partial sets in \( T' \).

The main idea of the algorithm is as follows. We first find an arbitrary maximal f-r-set packing \( T_0 \). This can be done in polynomial time. If \( |T_0| \geq R \), we are done. Otherwise, \( T_0 \) has less than \( R \) \( r \)-subsets. Then, we extend all of the possible valid partial r-set packings \( T_R \) with the form \( \{\sigma_1 = (c_1, *, *), \sigma_2 = (c_2, *, *), \ldots, \sigma_R = (c_R, *, *)\} \) to a solution, where each \( c_i \) belongs to \( S(T_0) \). To extend each of them, we first make a copy \( Q_R \) of \( T_R \) to calculate and store some temporary dates, and then replace each partial set \( \sigma^* \) in \( Q_R \) by a set, which \( \sigma^* \) is consist with, in \( V \) such that the replacement would not make \( Q_R \) invalid. If all partial sets of \( Q_R \) can be replaced by this way, we are done. Otherwise, we extend \( T_R \) by trying all possibilities that replace a * of a certain partial set \( \sigma^* \) in \( T_R \) by an element \( c \in S(Q_R) \) such that each of the replacements guarantees that \( T_R \) is valid and \( \sigma^* \) is consistent with at least one set of \( V \) after the replacement. Clearly, if the given instance is a true-instance, there must be at least one of the possibilities that leads to a solution. After each extension of \( T_R \), we reset \( Q_R = T_R \) and continue to extend \( T_R \) for each possibility by the same way. We describe the FPT algorithm for MrSP in Algorithm 1.

The time complexity is easy to see: each possible replacement corresponds to a branching. Since the size of \( S(Q_R) \) is always bounded by \( r \cdot R \), each extension can have at most \( r \cdot R \) branchings. Since each extension replaces one * in \( T_R \) and there are at most \( (r - 1) \cdot R \) *’s, the depth of the search tree is bounded by \( (r - 1) \cdot R \), implying that the time complexity of the algorithm is \( O^*((r \cdot R)^{(r-1) \cdot R}) \).

In the following, we show how to reduce r-AV to MrSP. Let \( E = ((C \cup \{p\}, \Pi_Y), \Pi_T, R) \) be an instance of r-AV. Let \( \Pi_p \subseteq \Pi_T \) be the multiset of all votes approving \( p \) in \( \Pi_T \) and \( \Pi_p = \Pi_T \cap \Pi_p \). The following observations is clearly true.

**Observation 12** If \( R > |\Pi_p| \), then \( E \) is a true-instance if and only if \( p \) is the unique winner in the election \( (C \cup \{p\}, \Pi_Y \cup \Pi_p) \).

Due to the above observation, we can solve the problem in the case that \( R > |\Pi_p| \); if \( \Pi_p \) is a solution return “Yes”, otherwise, return “No”. The following observations are useful to deal with the remaining case.

**Observation 13** No solution of r-AV contains a vote approving a candidate \( c \) with \( SC_Y (c) \geq SC_Y (p) + R - 1 \).

**Observation 14** If \( R \leq |\Pi_p| \) and \( E \) is a true-instance, then there must be a solution containing exactly \( R \) votes from \( \Pi_p \) but none vote from \( \Pi_p \).
Algorithm 1: \textit{FPTT} algorithm for MrSP

1. find an arbitrary maximal $f$-$r$-set packing $T_0$ in polynomial time;
2. if $|T_0| \geq R$ then
   3. return Yes;
5. \textbf{if} there is no valid ordering over $S(T_0)$ then /* The correctness here is based on the observation that for any solution of a true-instance, any set in the solution must intersect with $S(T_0)$. Otherwise, $T_0$ cannot be a maximal $f$-$r$-set packing since we can add the set in the solution which does not intersect $S(T_0)$ to $T_0$, contradicting that $T_0$ is maximal. */
   6. return No;
7. \textbf{end}
8. \textbf{forall the possible valid ordering} $(c_1, c_2, \ldots, c_R)$ over $S(T_0)$ \textbf{do}
   9. let $T_R = \{\sigma_1^*, \sigma_2^*, \ldots, \sigma_R^*\}$, where the partial set $\sigma_i^*$ contains the element $c_i$ and two “*” symbols;
10. Expand($T_R$);
11. \textbf{end}
12. Return No;

Procedure Expand($T_R$)

1. $Q_R = T_R$;
2. \textbf{foreach r-subset $\sigma$ in $V$ do}
3. \textbf{if} $\exists \sigma^*$ in $Q_R$ with $\text{reg}(\sigma^*) \subseteq \sigma$ and replacing $\sigma^*$ by $\sigma$ will not make $Q_R$ invalid then
4. \hspace{1cm} $Q_R = (Q_R \ominus \{\sigma^*\} \uplus \{\sigma\})$
5. \textbf{end}
6. \textbf{end}
7. \textbf{if} $Q_R$ contains no partial set then
8. \hspace{1cm} return Yes;
9. \textbf{end}
10. \textbf{if} no $\sigma^*$ was replace by some $\sigma$ in Step 4 then
11. \hspace{1cm} terminate;
12. \textbf{end}
13. \textbf{pick a partial set $\sigma^*$ in $Q_R$;}
14. \textbf{if} $\sigma^*$ is the only partial set in $Q_R$ then
15. \hspace{1cm} \textbf{if} there is a $\sigma \in V$ such that $Q_R \ominus \{\sigma^*\} \uplus \{\sigma\}$ is a $f$-$r$-set packing then
16. \hspace{2cm} return Yes
17. \hspace{2cm} else
18. \hspace{3cm} terminate;
19. \hspace{2cm} end
20. \textbf{end}
21. \textbf{forall the $c$ in $S(Q_R)$ do}
22. \textbf{if} replacing a “*” in $\sigma^*$ by $c$ gives a partial set which is consist with at least one set in $V$ and $T_R \ominus \{\sigma^*\} \uplus \{\sigma^* \ominus \{\ast\} \uplus \{c\}\}$ is valid then
23. \hspace{1cm} $T_R = T_R \ominus \{\sigma^*\} \uplus \{\sigma^* \ominus \{\ast\} \uplus \{c\}\}$;
24. \hspace{1cm} Expand($T_R$);
25. \textbf{end}
26. \textbf{end}
Due to Observation 13, we can safely remove all votes that approve some candidate $c$ with $SC_V(c) \geq SC_V(p) + R - 1$ in $\Pi_T$. Based on Observation 14, we reduce the problem to MrSP as follows: the universal set $C = C \setminus C'$ where $C'$ is the set of candidates with $SC_V(c) \geq SC_V(p) + R - 1$; the mapping $f$ is defined as $f(c) = SC_V(p) + R - SC_V(c) - 1$; the multiset $V$ contains exactly $n$ copies of $\{c_i, c_j, c_k\}$ for all $c_i, c_j, c_k \in C \setminus C'$ where there are exactly $n$ votes approving $c_i, c_j, c_k$ and $p$ in $\Pi_T$.

5 Conclusion

In this paper, we study a $k$-peaked election model which is a natural generalization of the single-peaked model in a way that at most $k$ peaks are allowed in each vote of the election. Moreover, we study complexities of many SP-AV control problems. We found that some problems such as $r$-AV which is $\mathcal{NP}$-hard in general and polynomial time solvable in single-peaked elections are polynomial time solvable under 2-peaked elections, where $r$ is a constant. However, most $\mathcal{NP}$-hardness of the SP-AV control problems hold in 2, or 3-peaked elections, in contrast to the polynomial time solvable in single-peaked elections. Our main results are showed in Table 1.

Besides, we study $r$-AV and $r$-DV from the viewpoint of parameterized complexity and proved both problems are $\mathcal{FP}$ with respect to the number of added votes and deleted votes, respectively, where $r$ is a constant.

Other further research direction could be studying more strategic behaviors such as manipulation and bribery for further voting protocols under $k$-peaked elections for different values of $k$.

6 Acknowledgement

The author sincerely thanks the anonymous referee(s) who have made several corrections and suggestions, which helped in improving the paper.

References

1. K. J. Arrow. A difficulty in the concept of social welfare. Journal of Political Economy, 58(4):328–346, 1950.
2. J. J. Bartholdi, C. A. Tovey, and M. A. Trick. How hard is it to control an election. In Mathematical and Computer Modeling, pages 27–40, 1992.
3. D. Baumeister, G. Erdélyi, E. Hemaspaanda, L. A. Hemaspaanda, and J. Rothe. chapter 10, pages 199–251. Handbook on Approval Voting. Springer Berlin Heidelberg, 2010.
4. 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.
5. N. Betzler, A. Slinko, and J. Uhlmann. On the computation of fully proportional representation. manuscript, 2011.
6. D. Black. On the rationable of group decision-making. Journal of Political Economy, 56:23–34, 1948.
7. 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.
8. F. Brandt, M. Brill, E. Hemaspaanda, and L. A. Hemaspaanda. Bypassing combinatorial protections: Polynomial-time algorithms for single-peaked electorates. In AAAI, pages 715–722, 2010.
9. D. Cornaz, L. Galand, and O. Spanjaard. Bounded single-peaked width and proportional representation. In ECAI, pages 270–275, 2012.
10. G. Demange. Single-peaked orders on a tree. Mathematical Social Sciences, 3(3):389–396, 1983.
11. R. G. Downey and M. R. Fellows. Parameterized Complexity. Springer, 1999.
12. G. Erdélyi, M. Lackner, and A. Pfandler. The complexity of nearly single-peaked consistency. In COMSOC, 2012.
13. G. Erdélyi, M. Nowak, and J. Rothe. Sincere-strategy preference-based approval voting broadly resists control. In MFCS, pages 311–322, 2008.
14. B. Escoffier, J. Lang, and M. Öztürk. Single-peaked consistency and its complexity. In ECAI, pages 366–370, 2008.
15. P. Faliszewski, E. Hemaspaanda, and L. A. Hemaspaanda. The complexity of manipulative attacks in nearly single-peaked electorates. In TARK, pages 228–237, 2011.
16. P. Faliszewski, E. Hemaspaanda, L. A. Hemaspaanda, 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.
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. J. R. Griggs and D. B. West. Extremal values of the interval number of a graph. *SIAM Journal on Algebraic and Discrete Methods*, 1(1):1–7, 1980.
19. J. Guo and R. Niedermeier. Invitation to data reduction and problem kernelization. *SIGACT News*, 38(1):31–45, 2007.
20. 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.
21. W. Jia, C. Zhang, and J. Chen. An efficient parameterized algorithm for m-set packing. *J. Algorithms*, 50(1):106–117, 2004.
22. J. Mark Keil. On the complexity of scheduling tasks with discrete starting times. *Operations Research Letters*, 12(5):293–295, 1992.
23. A. Lin. The complexity of manipulating k-approval elections. In ICARTT (2), pages 212–218, 2011.
24. 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.
25. N. Megiddo. On the complexity of solving the generalized set packing problem approximately. Technical report, IBM Almaden Research Center, 1987.
26. R. Niedermeier. *Invitation to Fixed-parameter Algorithms*. Oxford University Press Inc, 2006.
27. T. Walsh. Uncertainty in preference elicitation and aggregation. In AAAI, pages 3–8, 2007.
Appendices

A  Proof of Lemma 1

To prove Lemma 1, we prove the following claim first.

An interval is a *trivial interval* if this interval has only one element. For a graph \( G \), an interval representation \( \mathcal{I}(G) \) of \( G \) and a vertex \( u \in V(G) \), we call intervals in \( \mathcal{I}_u \) *u-intervals*. We say an interval \( I \) is a *left-most (resp. right-most) interval* if there is no other interval \( I' \) with \( l(I') < l(I) \) (resp. \( r(I') > r(I) \)). For a vertex \( u \in V(G) \), we use \( G \setminus \{u\} \) to denote the graph obtained from \( G \) by removing the vertex \( u \). For two graphs \( G := (V,E) \) and \( G' := (V',E') \), we denote by \( G \cup G' \) the graph \( (V \cup V', E \cup E') \).

**Claim:** For every graph \( G = (V, E) \) with bounded degree-3 and every \( v \in V \), there is a 2-interval representation \( \mathcal{I}(G,v) \) of \( G \) such that:

1. the left-most or the right-most interval is a \( v \)-interval;
2. each degree-(\( \leq 1 \)) vertex is represented by a trivial interval;
3. each degree-2 vertex \( w \) is represented by two trivial intervals or a closed interval \([x_1, x_2]\) with \( x_1 < x_2 \) such that no \( w \)-interval intersect \((x_1, x_2)\) for \( u \in V(G) \setminus \{w\} \);
4. for each degree-3 vertex \( w \in V \), \( I_w := \{I_w^1, I_w^2\} \) is one of the following kinds:
   (a) \( I_w^1 := [x_1, x_1], I_w^2 := [x_2, x_3], x_1 < x_2 < x_3 \) and \( \forall u \in V(G) \setminus \{w\} \) such that \( r(I(u)) \in (x_2, x_3) \) or \( l(I(u)) \in (x_1, x_2) \);
   (b) \( I_w^1 := [x_1, x_2], I_w^2 := [x_3, x_3], x_1 < x_2 < x_3 \) and \( \forall u \in V(G) \setminus \{w\} \) such that \( r(I(u)) \in (x_1, x_2) \) or \( l(I(u)) \in (x_2, x_3) \).

where \( I(u) \) is an \( u \)-interval.

**Proof.** We prove the claim by induction on \( n := V(G) \). The claim is clearly true for \( n = 1 \). Suppose that the claim is true for all graphs \( G \) with less than \( n \) vertices. Consider a graph \( G \) having \( n \) vertices. Without loss of generality, assume that \( G \) is connected (our proof is a construction proof, thus, if the given graph is disconnected, we can prove for each components, and then prove for \( G \) by combining all 2-representations of the components without any intersection). Let \( v \) be any fixed vertex of \( G \). If \( v \) is a degree-3 vertex with \( N_G(v) = \{a, b, c\} \). We consider the following subcases.

**Case 1.** \( v \) dose not belong to any circle.

Let \( C_a, C_b \) and \( C_c \) be the three components containing \( a, b \) and \( c \) in \( G \setminus \{v\} \), respectively. By induction, we know that \( C_a, C_b \) and \( C_c \) have 2-interval representations \( \mathcal{I}(C_a, a) \), \( \mathcal{I}(C_b, b) \) and \( \mathcal{I}(C_c, c) \), respectively, which satisfy all conditions in the above claim. Consider the following cases:

**Case 1.1** All \( a, b, c \) are degree-2 vertices in \( G \setminus \{v\} \). Due to the induction, we have a 2-interval representation of \( C_a \cup C_b \cup C_c \) as:

\[
\begin{align*}
\mathcal{I}(C_a) & \quad \cdots \quad \mathcal{I}(C_b) & \quad \cdots \quad \mathcal{I}(C_c) \\
I(a) & \quad \cdots \quad I(b) & \quad \cdots \quad I(c)
\end{align*}
\]

Then, we can get a 2-interval representation satisfying all conditions of the claim for \( G \) by the following way (in all following figures, we ignore the real line).

\[
\begin{align*}
\mathcal{I}(G) & \quad \mathcal{I}(C_a) & \quad \mathcal{I}(C_b) & \quad \mathcal{I}(C_c) \\
I_v^1 & \quad I_v^2 & \quad I_v^1 & \quad I_v^2
\end{align*}
\]
For simplicity, in the following cases, we describe only the difference to Case 1.1. We only show the precondition of each case, and give a 2-interval representation satisfying all conditions in the lemma for $G$, by combining the 2-interval representations of some components of $G$.

**Case 1.2** Two of $a, b, c$ have degree two and the other one has degree one in $G \setminus \{v\}$. Without loss of generality, assume $a$ is the degree-1 vertex in $G \setminus \{v\}$ among $a, b, c$.

**Case 1.3** Two of $a, b, c$ have degree two and the other one has degree zero in $G \setminus \{v\}$. Without loss of generality, assume that $a$ is the degree-0 vertex in $G \setminus \{v\}$ among $a, b, c$.

**Case 1.4** One of $a, b, c$ has degree two and the others have degree one in $G \setminus \{v\}$. Without loss of generality, assume that $a$ and $b$ are the degree-1 vertices in $G \setminus \{v\}$ among $a, b, c$.

**Case 1.5** All $a, b, c$ have different degrees in $G \setminus \{v\}$. Without loss of generality, assume that $a, b$ and $c$ are degree-0, degree-1 and degree-2 vertices, respectively, in $G \setminus \{v\}$.

**Case 1.6** One of $a, b, c$ has degree two and others have degree zero in $G \setminus \{v\}$. Without loss of generality, assume that $a$ and $b$ are the degree-0 vertices in $G \setminus \{v\}$.

**Case 1.7** All $a, b, c$ have degree one in $G \setminus \{v\}$.

**Case 1.8** Two of $a, b, c$ have degree one and the other one has degree zero in $G \setminus \{v\}$. Without loss of generality, assume that $a$ be the degree-0 vertex in $G \setminus \{v\}$.
Case 1.9 one of $a, b, c$ has degree one and the others have degree zero in $G \setminus \{v\}$. Without loss of generality, assume that $a$ and $b$ are the degree-0 vertices in $G \setminus \{v\}$.

Case 1.10 all $a, b, c$ have degree zero in $G \setminus \{v\}$. This case is trivial since the graph now is a star with four vertices.

Now, we finish all subcases that $v$ is not in a circle. We consider the second case where $c$ is contained in a circle in the following.

Case 2. $v$ is in a circle $O := (v, a, u_1, u_2, \ldots, u_t, b, v)$.

We construct a 2-interval representation $\mathcal{I}(O)$ for $O$ as follows.

Let $G'$ be the graph obtained from $G$ by deleting $v$ and all edges in the circle $O$. By the induction, there is a 2-interval representation of $G'$ which satisfy all conditions of the claim. Let $\mathcal{I}(G', c)$ be such a 2-interval representation with a $c$-interval as the left-most interval. Then, by the similar method used above we can combine $\mathcal{I}(O)$ and $\mathcal{I}(G')$ to derive a 2-interval representation of $G$ satisfying all conditions of the claim. We have three subcases here.

Case 2.1 $c$ is a degree-2 vertex in $G'$

Case 2.2 $c$ is a degree-1 vertex in $G'$.

Case 2.3 $c$ is an isolated vertex in $G'$. 

19
The above figures illustrate the case that $v$ is a degree-3 vertex. In the following, we consider the case that $v$ is a degree-2 vertex with $N_G(v) = \{a, b\}$. We consider the following cases.

**Case 1.** $v$ dose not belong to any circle.

Let $C_a$ and $C_b$ be the components containing $a$ and $b$ in $G \setminus \{v\}$, respectively. By induction, we know that $C_a$ and $C_b$ have 2-interval representations $I(C_a, a)$ and $I(C_b, b)$ which satisfy all conditions in the above claim, respectively. Consider the following subcases:

**Case 1.1** both $a$ and $b$ are degree-2 vertex in $G \setminus \{v\}$.

**Case 1.2** one of $a$ and $b$ has degree one and the other one has degree two in $G \setminus \{v\}$. Without loss of generality, assume that $a$ is the degree-1 vertex in $G \setminus \{v\}$.

**Case 1.3** both $a$ and $b$ are degree one vertices in $G \setminus \{v\}$.

**Case 1.4** one of $a$ and $b$ has degree zero and the other one has degree one in $G \setminus \{v\}$. Without loss of generality, assume that $a$ is the degree-0 vertex in $G \setminus \{v\}$.

**Case 1.5** both $a$ and $b$ are isolated vertices in $G \setminus \{v\}$. This case is trivial since the graph now contains only three vertices and two edges.

**Case 2.** $v$ is in some circle $O := (v, a, u_1, u_2, \ldots, u_t, b, v)$.

Let $I(O)$, $G'$ and $I(G', c')$ where $c'$ be any vertex in $V \setminus \{v, a, b\}$, are defined as above. Then, it is easy to verify that a 2-interval representation of $G$ which satisfy all conditions of the claim can be constructed by just combine $I(O)$ and $I(G', c')$ without any intersection.

Now we consider the last case that $v$ is a degree one vertex in $G$ with $N_G(v) = \{a\}$. Let $G'$ be the graph obtained from $G$ by removing $v$. The following pictures show the cases when $a$ is a degree-2 and degree-1 vertex in $G'$, respectively (from left to right).

The proof of the above claim directly gives a polynomial-time algorithm for construction a 2-interval representation satisfying all conditions in the claim for any graph $G$ with bounded degree-3.

Given the correctness of the above claim, it is easy to see that a 2-interval representation which satisfy all conditions of Lemma 1 for graph $G$ with bounded degree-3 can be found in polynomial-time by adding some dummy intervals in $I(G, v)$ for degree-(≤ 2) vertices, where $I(G, v)$ is the 2-interval representation constructed as in the proof of the claim.