ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-277

Complexity of Control by Partitioning Veto and Maximin Elections and of Control by Adding Candidates to Plurality Elections

Cynthia Maushagen[1] and **Jörg Rothe**[1]

Abstract. Control by partition refers to situations where an election chair seeks to influence the outcome of an election by partitioning either the candidates or the voters into two groups, thus creating two first-round subelections that determine who will take part in a final round. The model of partition-of-voters control attacks is remotely related to "gerrymandering" (maliciously resizing election districts). While the complexity of control by partition (and other control actions) has been studied thoroughly for many voting systems, there are no such results known for the important veto and maximin voting systems. We settle the complexity of control by partition for veto in a broad variety of models and for maximin with respect to destructive control by partition of candidates. We also observe that a reduction from the literature [8] showing the parameterized complexity of control by adding candidates to plurality elections, parameterized by the number of voters, is technically flawed by giving a counterexample, and we show how this reduction can be fixed.

1 INTRODUCTION

Along with manipulation [2, 9] and bribery [15, 17], electoral control [3, 23] has been the focus of much attention in computational social choice; see the book chapters by Faliszewski and Rothe [18] and Baumeister and Rothe [5] for a survey of the related results. Control scenarios model settings where an external agent, commonly referred to as the *chair*, seeks to influence the outcome of an election by such actions as adding, deleting, or partitioning either the candidates or the voters. We here focus on control by partition.

The above-mentioned chapters and the papers cited therein comprehensively describe applications of voting in artificial intelligence, multiagent systems, ranking algorithms, meta-websearch, etc., and they discuss how computational complexity can be used to provide some protection against manipulation, bribery, and control attacks. In particular, they give real-world examples of the various control types introduced by Bartholdi et al. [3] for the constructive control goal where the chair aims at making a given candidate win and by Hemaspaandra et al. [23] for destructive control where the goal is to prevent a given candidate's victory.

The complexity of control has been studied for many voting systems, including plurality, Condorcet, and approval voting [3, 23, 7] and its variants [14, 12], Copeland [17, 7], Borda [34, 11, 28, 8], (normalized) range voting [29], and Schulze voting [32, 30] (see the book chapters [18, 5] for an overview). Comparing veto (a.k.a. antiplurality) and plurality, even though these two scoring protocols are defined in similarly simple way, they behave quite differently for constructive coalitional weighted manipulation: While this problem is easy to solve in plurality for any number of candidates, it is NP-complete in veto for three or more candidates [9].[2] The main motivation of this paper is to find out whether veto similarly parts company from plurality regarding the complexity of control.

Perhaps a bit surprisingly, the important voting systems veto and maximin have not been investigated in terms of their control complexity by partition of either candidates or voters but only with respect to control by adding or deleting candidates or voters: Faliszewski et al. [16] studied maximin and Lin [26] studied veto for these control types in terms of their classical complexity, and their parameterized complexity has been explored by Liu and Zhu [27] for maximin and by Chen et al. [8] for veto and maximin. To the best of our knowledge, complexity results for control by partition have been missing for these two systems to date. This is all the more surprising as control by partition of voters provides a simplified model of gerrymandering (i.e., maliciously resizing election districts), a particularly natural control type known from the real world. One reason why these control scenarios have been neglected so far for veto and maximin may be that proofs for control by partition tend to be technical and challenging. We settle the complexity of control by partition for veto in a broad variety of models and for maximin with respect to destructive control by partition of candidates.

Since most of the known results on control by partition are in the original model as suggested by Bartholdi et al. [3] where the candidates or voters can be partitioned into two sets of arbitrary sizes, we will focus on this model too, in order to allow for comparability of results. However, we suggest to also study these problems for veto and maximin in the more refined models due to Erdélyi et al. [13] that restrict such partitions to sets of roughly the same size and due to Puppe and Tasnádi [33] that take geographical constraints into account when resizing election districts. Note that Bachrach et al. [1] study a related but different aspect of misrepresentation in district voting: Their "misrepresentation ratio" quantifies the deviation from proportional representation in district-based elections and they prove bounds on this ratio for various voting rules including veto.

In addition, we observe that a reduction due to Chen et al. [8, Theorem 1] showing the parameterized complexity of constructive control by adding candidates to plurality elections, where the parameter

[1] Institut für Informatik, Heinrich-Heine-Universität Düsseldorf, 40225 Düsseldorf, Germany, email: Cynthia.Maushagen@uni-duesseldorf.de, rothe@cs.uni-duesseldorf.de

[2] Indeed, Hemaspaandra and Hemaspaandra [20] proved a dichotomy result saying that plurality is the only nontrivial scoring protocol for which constructive coalitional weighted manipulation is easy (and Conitzer et al. [9] observed this too for the case of three candidates).

is the number of voters, is technically flawed. Specifically, we give a counterexample showing that their reduction maps a no-instance of the problem MULTI-COLORED-CLIQUE to a yes-instance of this control problem, and we show how this reduction can be fixed.

2 PRELIMINARIES

In this section, we define the needed voting systems and control problems and give some background on computational complexity.

2.1 Elections, Plurality, Veto, and Maximin Voting

An election is given by a pair (C,V), where C is a set of candidates and V a list of the voters' preferences over the candidates (which we will simply refer to as votes). We will consider only preferences that are linear orders (strict rankings) with the left-most candidate being the most preferred one. For example, a preference $d\ c\ a\ b$ means that this voter prefers d to c, c to a, and a to b.

We will consider three well-known voting systems: plurality, veto (a.k.a. antiplurality), and maximin (a.k.a. Simpson). In *plurality*, every voter gives one point to her most preferred candidate, and whoever scores the most points wins. Plurality is the perhaps simplest and still very prominent positional scoring protocol, a class of important voting systems that are based on the candidates' positional scores. In *veto*, every voter vetoes her least preferred candidate, which means that this candidate gets no point while all other candidates receive one point from this voter, and whoever scores the most points wins. Veto is another prominent positional scoring protocol. By contrast, *maximin voting* is based on the pairwise comparisons between the candidates and belongs to the class of Condorcet-consistent voting rules.[3] Given an election (C,V), for any two candidates $c,d \in C$, let $N(c,d)$ denote the number of voters preferring c to d. The *maximin score of* c is $\min_{c \neq d} N(c,d)$, and whoever has the largest maximin score wins the election.

2.2 Control Problems

We consider control by partition of either candidates or voters, as defined by Bartholdi et al. [3] and—for destructive control—by Hemaspaandra et al. [23].[4] The definitions below have been used in many papers; we refer to the book chapters by Faliszewski and Rothe [18] and Baumeister and Rothe [5] for the formal definitions of all problems studied here and for real-world examples motivating each control scenario we are interested in. In each such control scenario, starting from a given election (C,V) and a distinguished candidate $c \in C$, we form two subelections—either (C_1,V) and (C_2,V) where C is partitioned into C_1 and C_2 (i.e., $C_1 \cap C_2 = \emptyset$ and $C_1 \cup C_2 = C$), or (C,V_1) and (C,V_2) where V is partitioned into V_1 and V_2 (i.e., $V_1 \cap V_2 = \emptyset$ and $V_1 \cup V_2 = V$)—whose winners move forward to a final round if they survive the given tie-handling rule: either *ties-eliminate* (TE) that requires that only unique winners of a first-round subelection move forward, or *ties-promote* (TP) that requires that all winners of a first-round subelection move forward.

[3] A *(weak) Condorcet winner* is a candidate who defeats (ties-or-defeats) every other candidate in pairwise comparison. Condorcet winners do not always exist, but when they do, they are unique, whereas it is possible that there are several weak Condorcet winners. A voting rule is *Condorcet-consistent* if it respects the Condorcet winner whenever one exists.

[4] Constructive control by adding candidates, also due to Bartholdi et al. [3], will be defined in Section 6 because this control type will be considered only there.

Such a partition of either C or V is the chair's control action, and the chair's goal is either to ensure that the distinguished candidate c wins the final round (in the *constructive* case) or to prevent c's victory (in the *destructive* case), where the final round is always held with all votes from V. In the case of candidate control, we further distinguish between *run-off partition of candidates*, where the winners of (C_1,V) and (C_2,V) surviving the tie-handling rule face each other in the final run-off, and *partition of candidates*, where the winners of (C_1,V) surviving the tie-handling rule face all candidates of C_2 in the final round.

For each such control scenario, we can define a decision problem. As an example, we formally define the decision problem associated with constructive control by partition of voters in model TE for some given voting system $\mathscr{E}$:

$\mathscr{E}$-CONSTRUCTIVE-CONTROL-BY-PARTITION-OF-VOTERS-TE	
Given:	An election (C,V) and a distinguished candidate $c \in C$.
Question:	Can V be partitioned into V_1 and V_2 such that c is the unique $\mathscr{E}$ winner of the two-round election where the winners of the two first-round subelections (C,V_1) and (C,V_2) who survive tie-handling rule TE run against each other in a final round (with the votes from V correspondingly restricted)?

The above problem (denoted by $\mathscr{E}$-CCPV-TE—the shorthands of the other problems to be used later on will be clear from this example) is defined in the *unique-winner model*. We will also consider the *nonunique-winner model* where the question is changed to ask whether c is a winner (possibly among several winners) of the final round, and we will always specify the winner model we are referring to.

For a control type $\mathfrak{C}$ (such as constructive control by partition of voters in model TE), an election system $\mathscr{E}$ is said to be *immune to* $\mathfrak{C}$ if it is impossible for the chair to reach her control goal (e.g., to make the given candidate c a unique winner in the constructive case for the unique-winner model, or to ensure that c is not a winner in the destructive case for the nonunique-winner model) via exerting control of type $\mathfrak{C}$; otherwise, $\mathscr{E}$ is said to be *susceptible to* $\mathfrak{C}$. It is easy to observe that the two voting systems we study here, veto and maximin, are susceptible to every type of control (in both winner models) we have defined above; due to space limitations we omit giving detailed examples verifying these claims. If an election system $\mathscr{E}$ is susceptible to some control type $\mathfrak{C}$, it is common to study the computational complexity of the associated control problem: We say $\mathscr{E}$ is *vulnerable to* $\mathfrak{C}$ if the control problem corresponding to $\mathfrak{C}$ can be solved in polynomial time, and we say $\mathscr{E}$ is *resistant to* $\mathfrak{C}$ if $\mathfrak{C}$ is NP-hard.

2.3 Computational Complexity

We assume that the reader is familiar with the basic notions of computational complexity, such as the complexity classes P (deterministic polynomial time) and NP (nondeterministic polynomial time) and with the notions of NP-hardness and NP-completeness, based on the polynomial-time many-one reducibility. For more background, we refer to the book by Garey and Johnson [19].

In Section 6, we will also be concerned with *parameterized* complexity. In particular, we consider a result about W[1]-hardness. W[1] is a parameterized complexity class that in some sense corresponds to the classical complexity class NP, and just as NP-hardness indicates that a problem is infeasible to solve in the sense of classical complexity theory (i.e., has no polynomial-time algorithm unless $\mathrm{P} = \mathrm{NP}$), W[1]-hardness can be taken as strong evidence that a problem is not even fixed-parameter tractable. For more background on parameterized complexity and fixed-parameter tractability, we refer to the books by Downey and Fellows [10] and Niedermeier [31].

3 CONTROL BY PARTITION OF VOTERS IN VETO ELECTIONS IN MODEL TE

In this section, we show that it is easy to control veto elections by partition of voters in model TE. We start with the constructive case.

3.1 Veto-CCPV-TE

We show that veto is vulnerable to constructive control by partition of voters in model TE, in both winner models. Essentially, the polynomial-time algorithm used to prove Theorem 1 exploits the fact that, due to the TE model, control is impossible only if either there are two candidates and the distinguished candidate is not already a veto winner (in the unique-winner model: is not already the only veto winner) of the given election, or there are more than two candidates and some candidate other than the distinguished candidate is not vetoed by any voter. In all other cases it is easy to find a successful partition that ensures the distinguished candidate's victory.

Theorem 1 *Veto-*CCPV-TE *is in* P *in both the unique-winner and the nonunique-winner model.*

Proof. The following polynomial-time algorithm solves the problem. Given an election (C,V) with n votes in V and a candidate $c \in C$, it proceeds as follows: (1) If there are no more than two candidates, then if c already is a winner (in the unique-winner model: the only winner) of (C,V), control is possible via the trivial partition $(V,\emptyset)$, so accept; otherwise, control is impossible, so reject. (2) Otherwise (i.e., if $|C| > 2$), if $score(d) = n$ for some $d \in C \setminus \{c\}$, control is impossible, so reject. (3) Otherwise (i.e., if $|C| > 2$ and $score(d) < n$ for all $d \in C \setminus \{c\}$), it is safe to accept, since control is possible via the partition (V_1,V_2) of V that puts all voters who veto c into V_1 and all other voters into V_2.

The above algorithm runs in polynomial time and is correct. This is obvious for step 1. Further, it is impossible for c to defeat the candidate d with $score(d) = n$ in step 2 (as d scores the maximum number of points in each first-round subelection, no matter how V is partitioned, which makes it impossible for c to win alone in any subelection). And in step 3, no candidate from V_1 can move to the final round, because either V_1 is empty (in case no one vetoes c) or each of the at least two candidates other than c wins subelection (C,V_1) with the same score and, therefore, will be eliminated in model TE. On the other hand, each candidate $d \neq c$ is vetoed by at least one voter ending up in V_2, whereas c is not vetoed by any voter in V_2 and thus wins subelection (C,V_2) and the final run-off. This argument applies to both the unique-winner and the nonunique-winner model. ❑

3.2 Veto-DCPV-TE

A similar algorithm works in the destructive case. Note that Theorem 2 follows immediately from Theorem 1 for the unique-winner model,[5] but not for the nonunique-winner model. Therefore, we present a proof (which in fact works for both winner models).

Theorem 2 *Veto-*DCPV-TE *is in* P *in both the unique-winner and the nonunique-winner model.*

Proof. Given an election (C,V) and a distinguished candidate c, our algorithm works as follows: (1) If $|C| = 1$, control is impossible, so reject. (2) If $|C| = 2$, determine the set of veto winners. If c wins alone, control is impossible, so reject. Otherwise, control is possible via the trivial partition $(V,\emptyset)$, so accept. (3) If $|C| > 2$, it is safe to outright accept, since control is always possible: Fix some candidate $d \neq c$ and partition V into (V_1,V_2) such that V_1 contains all voters vetoing d and V_2 contains all remaining voters.

The above algorithm obviously runs in polynomial time and its correctness is straightforward for steps 1 and 2, while it follows for step 3 from the observation that if either c or d is vetoed by everyone then (V_1,V_2) will be trivial (either $(\emptyset,V)$ or $(V,\emptyset)$) and will thus prevent c from winning, and if neither c nor d is vetoed by everyone then there is a candidate e, $c \neq e \neq d$, who ties for winner with c in (C,V_1), while d ties-or-defeats c in (C,V_2); in either case, c cannot move forward to the final round due to model TE. ❑

4 CONTROL BY PARTITION OF CANDIDATES IN VETO ELECTIONS

We now turn to control by partition of candidates in veto elections, considering both constructive and destructive control, both tie-handling models, TE and TP, both the unique-winner and the nonunique-winner model, and the partition problems both with and without run-off.

4.1 Veto-CCRPC-TE, Veto-CCRPC-TP, Veto-CCPC-TE, and Veto-CCPC-TP

We start by showing that veto is resistant to constructive control by run-off partition of candidates.

Theorem 3 *Veto-*CCRPC-TE *is* NP*-complete in both the unique-winner and the nonunique-winner model, and Veto-*CCRPC-TP *is* NP*-complete in the unique-winner model.*

Proof. Membership of Veto-CCRPC-TE in NP is obvious. To show that it is NP-hard, we reduce from ONE-IN-THREE-POSITIVE-3SAT, an adaption from the well-known NP-complete problem ONE-IN-THREE-3SAT where the clauses of the given boolean formula do not contain any negated variables [19, p. 259]:

	ONE-IN-THREE-POSITIVE-3SAT
Given:	A set X of boolean variables, a set S of clauses over X, each containing exactly three unnegated literals.
Question:	Does there exist a truth assignment to the variables in X such that exactly one literal is set to true for each clause in S?

[5] As noted by Hemaspaandra et al. [23, Footnote 5 on p. 257], for voting systems that always have at least one winner (such as veto), any destructive control problem in the unique-winner model disjunctively truth-table reduces to the corresponding constructive control problem in the nonunique-winner model.

Let (X,S) be an instance of ONE-IN-THREE-POSITIVE-3SAT with $X = \{x_1,\dots,x_m\}$ and $S = \{S_1,\dots,S_n\}$. Construct an election (C,V) with distinguished candidate $c \in C$ by defining $C = X \cup \{c,w\}$, where the elements of X from now on will also be viewed as candidates, and the list V of votes as follows:

# votes	preference	
$2n^2+1$:	$w\ c\ \cdots\ x_i$	for each $i \in \{1,\dots,m\}$
$n-1$:	$w\ \cdots\ c$	
1 :	$c\ \cdots\ w\ S_j \setminus \{x_i\}$	for each $j \in \{1,\dots,n\}$ and $x_i \in S_j$
$2n$:	$w\ \cdots\ c\ S_j$	for each $j \in \{1,\dots,n\}$

There are $m+2$ candidates and $(2n^2+1)m+2n^2+4n-1$ voters in the election. If a set of candidates occurs in such a vote, we tacitly assume a fixed ordering of its candidates in this preference. The dots in a vote represent all remaining candidates (in an arbitrary, fixed order). In particular, there are $3n$ votes of the form $c\ \cdots\ w\ S_j \setminus \{x_i\}$. If, say, clause S_1 contains the literals x_2, x_5, and x_7, then the corresponding three votes are

$$c\ \cdots\ w\, x_2\, x_5, \quad c\ \cdots\ w\, x_2\, x_7, \quad c\ \cdots\ w\, x_5\, x_7.$$

Candidate w alone wins in election (C,V), since the candidates score the following points:[6]

$$\begin{aligned} score(c) &= (2n^2+1)m+3n+2n^2, \\ score(w) &= (2n^2+1)m+3n+n-1+2n^2, \text{ and} \\ score(x_i) &\leq (2n^2+1)(m-1)+n-1+3n+2n^2. \end{aligned}$$

Obviously, the reduction can be computed in polynomial time. It remains to show that (X,S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT if and only if (C,V,c) is a yes-instance of Veto-CCRPC-TE (in both winner models).

($\Rightarrow$) If (X,S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT, then there is a subset $U = \{u_1,\dots,u_k\}$ of X (renaming its elements for convenience) such that $|U \cap S_j| = 1$ for each $j \in \{1,\dots,n\}$. We claim that partitioning C into $C_1 = U \cup \{c,w\}$ and $C_2 = C \setminus C_1$ ensures that c is the only veto winner (and thus, *a fortiori*, c is a veto winner). To see this, note that the candidates in subelection (C_1,V) have the following scores:

$$\begin{aligned} score(c) &= (2n^2+1)m+3n+2n^2, \\ score(w) &= (2n^2+1)m+n-1+2n+2n^2, \text{ and} \\ score(u_i) &\leq (2n^2+1)(m-1)+n-1+3n+2n^2. \end{aligned}$$

For c to win (C_1,V) alone, we have to show that $score(c) > score(w)$ and $score(c) > score(u_i)$ for all $u_i \in U$: First, $score(c) > score(w)$ is equivalent to $(2n^2+1)m+3n+2n^2 > (2n^2+1)m+n-1+2n+2n^2$, which in turn is equivalent to $3n > 3n-1$; second, $score(c) > score(u_i)$ is equivalent to $(2n^2+1)m+3n+2n^2 > (2n^2+1)(m-1)+n-1+3n+2n^2$, which in turn is equivalent to $2n^2+1 > n-1$.

Being the only veto winner of subelection (C_1,V), c will move forward to the final run-off. If more than one candidate wins subelection (C_2,V) (thus TE blocking them all from moving to the final run-off), c's overall victory is ensured. On the other hand, if some candidate $x_i \in C_2$ is the only veto winner of (C_2,V), c will face x_i in the run-off. However, since

$$score(c) \geq (2n^2+1)m+3n > n-1+2n^2 \geq score(x_i)$$

in the run-off $(\{c,x_i\},V)$, c wins the run-off and is the only overall veto winner. Thus (C,V,c) is a yes-instance of Veto-CCRPC-TE in the unique-winner model.

($\Leftarrow$) Conversely, let (X,S) be a no-instance of ONE-IN-THREE-POSITIVE-3SAT. Then, for each partition of X into X_1 and X_2, let k_i be the number of clauses containing i literals from X_1. We have $1 \leq k_0+k_1+k_2+k_3 \leq n$, since we started from a no-instance of ONE-IN-THREE-POSITIVE-3SAT. We will show that for each possible combination of the k_i (corresponding to each possible partition of X), candidate c cannot end up being a veto winner (*a fortiori*, c cannot be the only veto winner). Note that a partition of X induces a partition of $C = X \cup \{c,w\}$ into C_1 and $C_2 = C \setminus C_1$ (assuming, without loss of generality, that $c \in C_1$). It is enough to distinguish the three cases below, and in each case, we will show that c is not a veto winner.

Case 1: $C_1 = \{c,w\}$. Then $score(c) = 3n$ and $score(w) = (2n^2+1)m+n-1+2n^2 \geq 4n^2+n$, so w is the only veto winner of this subelection, and since c does not take part in the final run-off, c will not be an overall winner.

Case 2: C_1 contains c but not w. It is enough to show that w is the only winner of the other subelection, (C_2,V), since if c wins (C_1,V), then either c is not promoted to the final round due to TE (if there are other winners) or c loses the final round as we have seen in Case 1. In subelection (C_2,V), for each $x_i \in C_2$, we have

$$\begin{aligned} score(w) &\geq (2n^2+1)m+n-1+2n^2 \\ &> (2n^2+1)(m-1)+n-1+3n+2n^2 \\ &\geq score(x_i), \end{aligned}$$

where the "greater than" follows from $2n^2+1 > 3n$, which is true for all $n > 1$. (For $n = 1$, however, we would have started from a yes-instance of ONE-IN-THREE-POSITIVE-3SAT, which contradicts our assumption.) Thus w is the only veto winner of (C_2,V), which precludes c's overall victory in this case.

Case 3: C_1 contains c, w, and some elements of X. Distinguish the following three subcases.

Case 3.1: $k_0 \geq 2$. In this case, we have

$$\begin{aligned} score(c) &\leq (2n^2+1)m+3n+(n-k_0)2n \text{ and} \\ score(w) &\geq (2n^2+1)m+n-1+2n^2. \end{aligned}$$

Regardless of the points the elements of X in C_1 score, it suffices to show that $score(c) \leq score(w)$. This, however, holds since (for $k_0 \geq 2$) the inequality $2n+1 \leq 2k_0n$ implies

$$(2n^2+1)m+3n+(n-k_0)2n \leq (2n^2+1)m+n-1+2n^2.$$

Case 3.2: $k_0 = 1$. In this case, we have

$$\begin{aligned} score(c) &\leq (2n^2+1)m+3n+(n-k_0)2n \text{ and} \\ score(w) &\geq (2n^2+1)m+n-1+2(n-1)+2n^2. \end{aligned}$$

Now, the inequality $3 \leq 2n$ (which is true for $n > 1$; the case $n = 1$ can again be excluded) implies $score(c) \leq score(w)$ also in this case.

Case 3.3: $k_0 = 0$. Since we have a no-instance, at least one clause must contain at least two literals from X_1, so

$$\begin{aligned} score(c) &= (2n^2+1)m+3n+2n^2 \text{ and} \\ score(w) &\geq (2n^2+1)m+n-1+2n+1+2n^2. \end{aligned}$$

The term $2n+1$ in $score(w)$ is due to the third row in V. Every clause S_j contains at least one literal corresponding to a candidate

[6] Here and in the following, we omit a detailed argumentation of why certain candidates score a certain number of points in some election, due to space limitations and since these scores can be determined straightforwardly.

x_i in C_1, so w gains at least two points per clause. Since at least one clause contains at least two literals corresponding to candidates in C_1, w receives all three possible points for this clause, which explains the important additional point. Again, it is enough to show $score(c) \leq score(w)$. But this follows since $3n+2n^2 \leq 2n^2+3n$ implies

$$(2n^2+1)m+3n+2n^2 \leq (2n^2+1)m+n-1+2n+1+2n^2.$$

By model TE, c cannot move forward to the final round and thus cannot win the overall election. As we have shown that c is not a veto winner in any partition of the candidates, (C,V,c) is a no-instance of Veto-CCRPC-TE.

The proof (omitted here) that Veto-CCRPC-TP is NP-complete in the unique-winner model works by suitably adapting the above proof. ❑

A minor tweak in the construction of the previous proof (namely, by having n instead of $n-1$ votes of the form $w \cdots c$, all else being equal) works for showing NP-hardness of both Veto-CCPC-TE and Veto-CCPC-TP in the nonunique-winner model. Other minor changes work in the unique-winner case. The proof of Theorem 4 is omitted due to space limitations.

Theorem 4 *Veto-*CCPC-TP *and Veto-*CCPC-TE *are* NP-*complete in both the nonunique-winner and the unique-winner model.*

4.2 Veto-DCRPC-TE and Veto-DCPC-TE

Now we turn to the destructive variant of the previous problem, but now in both winner models. We again show resistance via a reduction from ONE-IN-THREE-POSITIVE-3SAT.

Theorem 5 *Veto-*DCRPC-TE *is* NP-*complete in both the unique-winner and the nonunique-winner model.*

Proof. Membership of both problems in NP is again obvious. For showing NP-hardness, let (X,S) be an instance of ONE-IN-THREE-POSITIVE-3SAT with $X=\{x_1,\dots,x_m\}$ and $S=\{S_1,\dots,S_n\}$. Construct an election (C,V) with $C=X\cup\{c,w\}$, $c\in C$ being the distinguished candidate, and the following list of votes:

# votes	preference	
$3n+1:$	$c\ w\ \cdots\ x_i$	for each $i\in\{1,\dots,m\}$
$2n+2:$	$c\ \cdots\ w\ S_j$	for each $j\in\{1,\dots,n\}$
$n:$	$c\ \cdots\ w$	
$1:$	$w\ \cdots\ c\ S_j\setminus\{x_i\}$	for each $j\in\{1,\dots,n\}$ and $x_i\in S_j$

The election contains $m+2$ candidates and $(3n+1)m+(2n+6)n$ voters. The reduction can be computed in polynomial time. It is easy to see that c is the only veto winner of election (C,V):

$$\begin{aligned} score(c) &= (3n+1)m+(2n+2)n+n+3n,\\ score(w) &= (3n+1)m+(2n+2)n+3n, \text{ and}\\ score(x_i) &\leq (3n+1)(m-1)+(2n+2)n+n+3n. \end{aligned}$$

We claim that (X,S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT if and only if (C,V,c) is a yes-instance of Veto-DCRPC-TE (in both winner models).

($\Rightarrow$) If (X,S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT, then there is a subset U of X such that $|U\cap S_j|=1$ for each $j\in\{1,\dots,n\}$. Partitioning C into $C_1=U\cup\{c,w\}$ and $C_2=C\setminus C_1$ ensures that c is not a veto winner (*a fortiori*, c is not the only veto winner), since c and w have the same score in subelection (C_1,V):

$$\begin{aligned} score(c) &= (3n+1)m+(2n+2)n+n+2n \text{ and}\\ score(w) &= (3n+1)m+(2n+2)n+3n, \end{aligned}$$

so, by model TE, c cannot move forward to the final round.

($\Leftarrow$) Conversely, let (X,S) be a no-instance of ONE-IN-THREE-POSITIVE-3SAT. As in the proof of Theorem 3, we consider all possible partitions of C into C_1 and C_2 (again assuming, without loss of generality, that $c\in C_1$) and show that c always is the only veto winner (*a fortiori*, c is a veto winner) overall.

Case 1: $C_1=\{c,w\}$. Then $score(c)=(3n+1)m+(2n+2)n+n$ and $score(w)=3n$, so c moves forward to the final round. If the other subelection, (C_2,V), has more than one winner, TE blocks them all, so c wins. If (C_2,V) has a unique winner, say x_i, we have $score(c)=(3n+1)m+(2n+2)n+n$ and $score(x_i)\leq 3n$ in the final round, $(\{c,x_i\},V)$, so c wins.

Case 2: C_1 contains c and some elements of X but not w.

$$\begin{aligned} score(c) &= (3n+1)m+(2n+2)n+n+3n \text{ and}\\ score(x_i) &\leq (3n+1)(m-1)+(2n+2)n+n+3n \end{aligned}$$

then imply that c scores at least $3n+1$ points more than any x_i and moves forward to the final round. If (C_2,V) has more than one winner, c outright wins; if either w or some x_i wins in (C_2,V), c wins the run-off as shown in Case 1.

Case 3: C_1 contains c, w, and some elements of X. Rename the elements of $U=C_1\cap X$ by $U=\{u_1,\dots u_\ell\}$. Let k be the number of clauses S_j such that $|S_j\cap U|=0$.

Case 3.1: $k>0$. Then the scores in (C_1,V) are:

$$\begin{aligned} score(c) &\geq (3n+1)m+(2n+2)n+n+2(n-k),\\ score(w) &= (3n+1)m+(2n+2)(n-k)+3n, \text{ and}\\ score(u_i) &\leq (3n+1)(m-1)+(2n+2)n+n+3n. \end{aligned}$$

For c to win subelection (C_1,V) alone, we need to show that $score(c)>score(w)$ and $score(c)>score(u_i)$ for each $u_i\in U$. Simplifying the scores of c and w, we get $2n^2+5n-2k>2n^2+5n-2nk-2k$, which is equivalent to $2nk>0$, which is true because $k>0$ and $n>0$. Obviously, c also wins out over each $u_i\in U$, since simplifying their scores yields $2n+1>2k$, which is true. In the run-off, c is either alone or faces some x_i (if x_i is the only veto winner of subelection (C_2,V)). By the argument just given, c triumphes over x_i and is the only overall veto winner.

Case 3.2: $k=0$. Since (X,S) is a no-instance, there is at least one clause S_j with $|S_j\cap U|\geq 2$ in this case. This implies the following scores in (C_1,V):

$$\begin{aligned} score(c) &\geq (3n+1)m+(2n+2)n+n+2n+1,\\ score(w) &= (3n+1)m+(2n+2)n+3n, \text{ and}\\ score(u_i) &\leq (3n+1)(m-1)+(2n+2)n+n+3n. \end{aligned}$$

Thus c is the only veto winner of subelection (C_1,V) and (by the above arguments) wins also the final run-off alone. Hence, (C,V,c) is a no-instance of Veto-DCRPC-TE. ❑

In both winner models, the problems DCRPC-TE and DCPC-TE are known to be identical for all voting systems [22, Thm. 8 on p. 386]; the proofs can be found in the related technical report by Hemaspaandra et al. [21]. Thus we have from Theorem 5:

Corollary 6 *Veto-*DCPC-TE *is* NP-*complete in both the unique-winner and the nonunique-winner model.*

4.3 Veto-DCRPC-TP and Veto-DCPC-TP

We next turn to the ties-promote model, TP. By slightly modifying the proof of Theorem 5, we will show resistance in both cases for the nonunique-winner model.

Theorem 7 *Veto-*DCRPC-TP *and Veto-*DCPC-TP *are* NP-*complete in the nonunique-winner model.*

Proof. Starting with Veto-DCRPC-TP, we only describe the differences with the construction given in the proof of Theorem 5. The only required change is that the votes of the form $c \cdots w$ (see the third row) occur $n-1$ instead of n times. The arguments showing the correctness of the construction then need to be adapted to model TP; the details are omitted here due to space limitations. Regarding Veto-DCPC-TP, note that DCRPC-TP and DCPC-TP are known to be identical problems in the nonunique-winner model for all voting systems [22, Thm. 8 on p. 386]. ❑

5 DESTRUCTIVE CONTROL BY PARTITION OF CANDIDATES IN MAXIMIN ELECTIONS

Finally, we turn to destructive control by partition of candidates in maximin elections. We start with the ties-eliminate model.

5.1 Maximin-DCRPC-TE and Maximin-DCPC-TE

While veto is vulnerable to both constructive and destructive control by partition of voters but not to the types of candidate control we have studied, maximin voting turns out to be vulnerable to destructive control by partition of candidates.

Theorem 8 *In both the unique-winner and the nonunique-winner model, Maximin-*DCRPC-TE *is in* P.

Proof. Given an election (C,V) with distinguished candidate $c \in C$ as input, our polynomial-time algorithm for Maximin-DCRPC-TE simply works as follows: If c is the Condorcet winner of (C,V), control is impossible, so reject; otherwise, accept.

To see that the algorithm is correct, note that control is always possible if c is not a Condorcet winner of (C,V): In the unique-winner model, we can argue that there is at least one candidate, say $d \in C$, such that $N(d,c) \geq N(c,d)$. Now, partitioning C into $C_1 = \{d\}$ and $C_2 = C \setminus C_1$ ensures that d moves forward to the final run-off, and even if c emerges as the only maximin winner of the other subelection, (C_2,V), and faces d in the run-off, c will not be the only maximin winner of the overall election. The proof in the nonunique-winner model is similar: If some candidate d defeats c, then again partition C into $C_1 = \{d\}$ and $C_2 = C \setminus C_1$, so d defeats c in the run-off. Otherwise, there must be a candidate e that ties-or-defeats c, so partitioning C into $C_1 = \{c,e\}$ and $C_2 = C \setminus C_1$ makes sure that due to the TE rule, c does not move forward to the run-off and does not win. On the other hand, if c is the Condorcet winner of (C,V), in both winner models, no partition of C can prevent c from being the only maximin winner of the overall election. ❑

Again, we can apply the known result that DCRPC-TE equals DCPC-TE for all voting systems [22, Thm. 8 on p. 386].

Corollary 9 *In both the unique-winner and the nonunique-winner model, Maximin-*DCPC-TE *is in* P.

5.2 Maximin-DCRPC-TP and Maximin-DCPC-TP

In the ties-promote model, TP, the algorithm used to prove Theorem 8 works as well, though the proof of correctness needs to be slightly adjusted. Note that, unlike in TE, DCRPC-TP and DCPC-TP are not known to coincide in the *unique*-winner model, though DCRPC-TP equals DCPC-TP in the *nonunique*-winner model [22, Thm. 8 on p. 386], as noted in the proof of Theorem 7.

Theorem 10 *In the unique-winner model, both Maximin-*DCRPC-TP *and Maximin-*DCPC-TP *are in* P.

Proof. Given an election (C,V) with distinguished candidate $c \in C$ as input, the simple polynomial-time algorithm for Maximin-DCRPC-TE from the proof of Theorem 8 also works here: If c is the Condorcet winner of (C,V), reject; otherwise, accept.

The proof of correctness is adjusted as follows. If c is the Condorcet winner of (C,V), our destructive goal can again never be reached: No partition of C can prevent c from being the only maximin winner of the overall election. On the other hand, if c is not a Condorcet winner of (C,V), we distinguish two cases: First, if c is a weak Condorcet winner of (C,V), there exists a candidate, say d, such that $N(d,c) = N(c,d)$; partitioning C into $C_1 = \{d\}$ and $C_2 = C \setminus C_1$ ensures that c will not be the only maximin winner of the overall election. Second, if c is not even a weak Condorcet winner of (C,V), there exists a candidate, say d, such that $N(d,c) > N(c,d)$; partitioning C into $C_1 = \{c,d\}$ and $C_2 = C \setminus C_1$ will ensure that c does not even win subelection (C_1,V). Obviously, this argument works both with and without run-off, i.e., both for Maximin-DCRPC-TP and Maximin-DCPC-TP. ❑

6 CONSTRUCTIVE CONTROL BY ADDING CANDIDATES IN PLURALITY ELECTIONS

In this section, we consider only a single control scenario (constructive control by adding candidates) for the simplest natural voting system, plurality. That plurality is resistant to this control type in the sense of Plurality-CCAC being NP-hard has already been known since the first paper on electoral control, due to Bartholdi et al. [3]. Recently, Chen et al. [8] considered the parameterized complexity of control problems for natural voting systems when there are only few voters. In particular, they proved that the parameterized variant of Plurality-CCAC, parameterized by the number of voters, is W[1]-hard by reducing from the W[1]-hard problem MULTI-COLORED-CLIQUE, parameterized by the clique order [8, Theorem 1].

However, while the proof sketch of this result does provide a very clever reduction, it is technically flawed. In this section, we first briefly present their reduction from the proof sketch of [8, Theorem 1], then give a counterexample showing that it is not correct, and finally fix this flaw by suitably adapting their reduction in order to make it correct. The W[1]-hard parameterized problem Chen et al. [8] reduce from is formally defined as follows.

MULTI-COLORED-CLIQUE	
Given:	An undirected graph $G = (V(G), E(G))$, where $V(G)$ is partitioned into h sets $V_1(G), \ldots, V_h(G)$ such that each $V_i(G) = \left\{v_1^{(i)}, \ldots, v_{n'}^{(i)}\right\}$ consists of exactly n' vertices with color i and G has only edges connecting vertices of distinct colors.
Parameter:	the number h of colors.
Question:	Does there exist a size-h clique containing some vertex for each color?

Given a voting rule $\mathscr{E}$, the parameterized problem $\mathscr{E}$-CCAC (parameterized by the number of voters) is defined as follows.

$\mathscr{E}$-CCAC	
Given:	A set C of registered candidates, a set A of as yet unregistered candidates, $C \cap A = \emptyset$, a list of preferences V over $C \cup A$, a nonnegative integer k, and a distinguished candidate $p \in C$.
Parameter:	the number of votes in V.
Question:	Does there exist a subset $A' \subseteq A$ such that $\|A'\| \leq k$ and p is an $\mathscr{E}$ winner of the election $(C \cup A', V')$ with V' being V restricted to $C \cup A'$?

We now describe the reduction from the proof sketch of Theorem 1 due to Chen et al. [8]. Let $G = (V(G), E(G))$ be a given undirected graph, where $V(G)$ is partitioned into h sets $V_1(G), \ldots, V_h(G)$ such that each $V_i(G) = \left\{v_1^{(i)}, \ldots, v_{n'}^{(i)}\right\}$ consists of exactly n' vertices with color i and G has only edges connecting vertices of distinct colors. Construct the following instance (C, A, V, k, p) of Plurality-CCAC:

- The set of registered candidates is $C = \{p, d\}$, where p is the distinguished candidate the chair wants to see win.
- The set A of unregistered candidates contains
  - a *vertex candidate* v for each $v \in V(G)$ and
  - two *edge candidates* (u, v) and (v, u) for each edge $\{u, v\} \in E(G)$.
- To specify the list V of votes, we adopt the following notation from [8]. Let $E(i, j)$ be the set of all edge candidates (u, v) with $u \in V_i(G)$ being colored i and $v \in V_j(G)$ being colored j.

  For each vertex $v_z^{(i)} \in V_i(G)$, let $L(v_z^{(i)}, j)$ be the set of all edge candidates $(v_z^{(i)}, v)$ with $v \in V_j(G)$ and $(v_z^{(i)}, v) \in E(G)$.

  For each i, j, $1 \leq i \neq j \leq n$, define the following two linear orders:

$$R(i,j) \quad : \quad v_1^{(i)}\, L(v_1^{(i)}, j) \, \cdots \, v_{n'}^{(i)}\, L(v_{n'}^{(i)}, j)$$
$$R'(i,j) \quad : \quad L(v_1^{(i)}, j)\, v_1^{(i)} \, \cdots \, L(v_{n'}^{(i)}, j)\, v_{n'}^{(i)}.$$

  Now we are ready to define the following three types of votes:

  1. For each i, there is one vote of the form $v_1^{(i)} \cdots v_{n'}^{(i)}\ d \cdots$.
  2. For each pair of colors i, j, $1 \leq i \neq j \leq n$, there are (a) $h-1$ votes of the form $E(i,j)\ d \cdots$, (b) one vote of the form $R(i,j)\ d \cdots$, and (c) one vote of the form $R'(i,j)\ d \cdots$.
  3. There are h votes of the form $d \cdots$ and h votes of the form $p \cdots$.
- At most $k = h + 2\binom{h}{2}$ candidates can be added.

Chen et al. [8] then argue that p can become a plurality winner by adding at most k candidates from A if and only if graph G has a size-h multi-colored clique (i.e., a clique containing a vertex for each color). However, we now present a counterexample for this claim.

Example 11 *Figure 1 shows a graph G corresponding to a no-instance of* MULTI-COLORED-CLIQUE*. In particular, the vertex set V(G) is partitioned into three sets containing two vertices each:*

$$V_1(G) = \{①, ⑥\}, \quad V_2(G) = \{❷, ❺\}, \quad V_3(G) = \{③, ④\}$$

but, obviously, G has no clique of size three.

However, we now show that the above construction maps this no-instance of MULTI-COLORED-CLIQUE *to a yes-instance of Plurality*-CCAC*. Indeed, from G we obtain the set $C = \{p, d\}$ of registered candidates, where p is the distinguished candidate the chair wants to see win. The set of unregistered candidates is*

$$A = \left\{ ①, ❷, ③, ④, ❺, ⑥, \overset{❷}{①}, \overset{①}{❷}, \overset{③}{❷}, \overset{❷}{③}, \overset{④}{❷}, \overset{❷}{④}, \overset{⑥}{④}, \overset{④}{⑥}, \overset{⑥}{❺}, \overset{❺}{⑥} \right\}$$

with six vertex candidates and ten edge candidates. Figure 2 gives the list V of votes over $C \cup A$, where the number before a vote indicates how many votes of this type there are according to the above construction. Finally, since $h = 3$, we are allowed to add $k = 3 + 2\binom{3}{2} = 9$ candidates. Since G has no clique of size three, it should be impossible to make p a plurality winner by adding at most $k = 9$ candidates. However, adding the candidates ⑥, ❷, ④, $\overset{❺}{⑥}$, $\overset{①}{❷}$, $\overset{④}{⑥}$, $\overset{③}{❷}$, $\overset{⑥}{④}$, $\overset{❷}{④}$ to the election implies that each candidate scores exactly three points. In particular, p has become a plurality winner, which shows that a no-instance of MULTI-COLORED-CLIQUE *has been mapped to a yes-instance of Plurality*-CCAC *by the reduction presented in the proof sketch of [8, Theorem 1].*

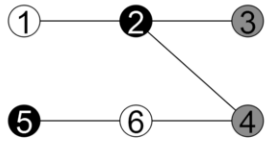

Figure 1: Counterexample for the reduction for [8, Theorem 1]

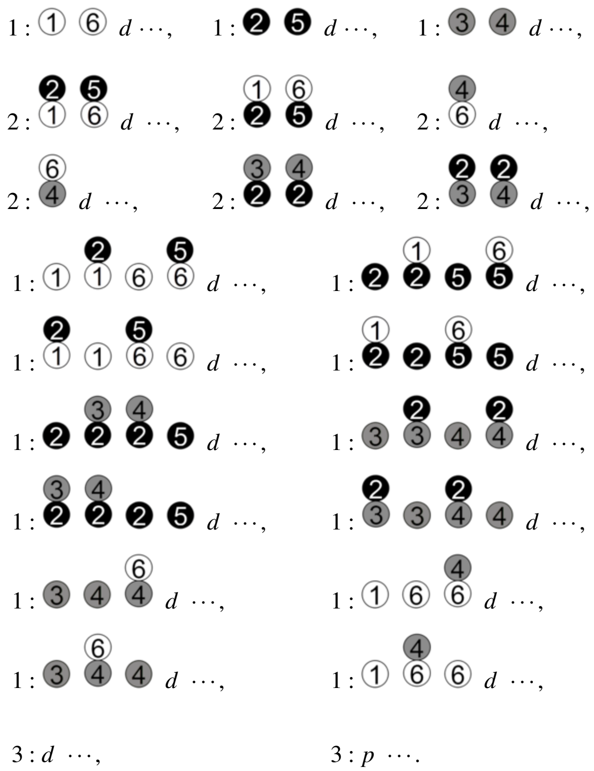

Figure 2: Constructing a yes-instance of Plurality-CCAC from a no-instance of MULTI-COLORED-CLIQUE according to the proof sketch of [8, Theorem 1]

Let us discuss the observation made in Example 11 in general and let us see how the reduction can be adapted so as to work correctly.

First note that p can never score more than h points, no matter how many candidates are added to the election. Thus, for p to be a winner, no other candidate must score more than h points. Candidate d will score at least h points, no matter if any (and how many) candidates are added. To prevent d from scoring more than h points, any size-$\left(h+2\binom{h}{2}\right)$ set $A' \subseteq A$ of added unregistered candidates must contain exactly one vertex candidate for each color and exactly one edge candidate of each (ordered) pair of colors. In their proof sketch Chen et al. [8, p. 2049] further say that "if A' contains two vertex candidates u, v but not the edge candidate (u,v) then, due to the orders $R(i,j) \succ d \succ \cdots$ and $R'(i,j) \succ d \succ \cdots$, either u or an edge candidate (u',v') (where $u' \in V_i(G)$, $v' \in V_j(G)$, but $(u',v') \neq (u,v)$) receives too many points, causing p not to win." [7] That is, they claim that adding two vertex candidates, $u \in V_i(G)$ and $v \in V_j(G)$, enforces addition of the edge candidate (u,v) because this would be required to ensure that the points from the two votes $R(i,j)\ d\ \cdots$ and $R'(i,j)\ d\ \cdots$ are scored by *different* candidates. This, however, is not always true. Indeed, to ensure that different candidates score points from these two votes, it is enough to add with u and v an edge candidate (u,v_j) such that v_j and v have the same color and $(u,v_j) \in E(G)$. Therefore, it in fact is possible to add two edge candidates (u,v) and (v',u') with $u,u' \in V_i(G)$, $v,v' \in V_j(G)$, $i \neq j$, and $(u,v) \neq (u',v')$.

In Example 11, the vertex candidates ❷ and ④ and also the edge candidate $\overset{③}{❷}$ have been added. The problem is that one is not forced to also add the edge candidate $\overset{❷}{③}$. The above argument is only correct if one assumes that the edge candidates (u,v) and (v,u) must always be added together. To enforce this, one can adapt the votes $E(i,j)\ d\ \cdots$ by requiring each edge candidate (u,v) is followed by the corresponding edge candidate (v,u). For instance, in Example 11 that means that the two votes $\overset{③}{❷}\ \overset{④}{❷}\ d\ \cdots$ are both changed to $\overset{③}{❷}\ \overset{❷}{③}\ \overset{④}{❷}\ \overset{❷}{④}\ d\ \cdots$. Now, if one adds two *unmatching* edge candidates (i.e., (u,v) and (v',u') with $(u,v) \neq (u',v')$, $u,u' \in V_i$, and $v,v' \in V_j$) then, without loss of generality, edge candidate (u,v) receives the points in the now modified votes $E(i,j)\ d\ \cdots = \cdots\ (u,v)\ (v,u)\ \cdots (u',v')\ (v',u')\ \cdots\ d\ \cdots$ and $E(j,i)\ d\ \cdots = \cdots\ (v,u)\ (u,v)\ \cdots (v',u')\ (u',v')\ \cdots\ d\ \cdots$. However, if one adds the *matching* edge candidates, say (u,v) and (v,u), then each of them receives a point only from one of these modified votes. This enforces that only *matching* edge candidates can be added (otherwise, p would not win). The votes $R\ (i,j) : v_1^{(i)}\ L(v_1^{(i)},j)\ \cdots\ v_{n'}^{(i)}\ L(v_{n'}^{(i)},j)$ and $R'(i,j) : L(v_1^{(i)},j)\ v_1^{(i)}\ \cdots\ L(v_{n'}^{(i)},j)\ v_{n'}^{(i)}$ then imply that with an edge candidate (u,v) also the candidate u must be added. If some other vertex candidate $u' \neq u$ were added, the above two votes restricted to candidates u' and (u,v) would either both be $u'\ (u,v)$ or both be $(u,v)\ u'$, which would give one of these two candidates too many points.

Obviously, matching vertex and edge candidates can be added only if there is a size-h multi-colored clique in the given graph G. If there is no such clique, at least one unmatching candidate has to be added.

7 CONCLUSIONS AND OPEN QUESTIONS

We have studied the complexity of control by partition of either voters or candidates for veto elections and of destructive control by partition of candidates for maximin. Recall that our main goal was to find out whether veto parts company from plurality regarding the complexity of control by partition. We have seen that, in stark contrast with constructive coalitional weighted manipulation where veto and plurality behave quite differently, the results obtained for control by partition in veto are exactly the same as those known for control by partition in plurality [3, 23]: Control by partition of candidates is hard, whereas control by partition of voters is easy. For veto, the complexity is still open for CCPV-TP and DCPV-TP and in some cases for one of the two winner models. Table 1 gives a detailed overview by comparing our results for veto with the known results for plurality due to Bartholdi et al. [3] and Hemaspaandra et al. [23] (which all were shown only in the unique-winner model), in particular indicating the open questions by question marks. In this table, V stands for vulnerability (i.e., the corresponding control problem is in P) and R for resistance (i.e., the corresponding control problem is NP-hard). By R* and $=^*$ (respectively, by R†) we indicate that this result has been shown only in the nonunique-winner (respectively, in the unique-winner) model (and, for our R* and R† entries, the question of whether these resistance results hold also in the other winner model is left open), while all other results hold in both the nonunique-winner and the unique-winner model. For maximin, we only obtained (easy) polynomial-time algorithms for destructive candidate control cases—which is similar to the results known for these control types in Copeland elections [17].

Control problem	Veto	Plurality
CCPV-TE	V (Thm. 1)	V
DCPV-TE	V (Thm. 2)	V
CCPV-TP	?	R
DCPV-TP	?	R
CCRPC-TE	R (Thm. 3)	R
DCRPC-TE = DCPC-TE	R (Thm. 5 and Cor. 6)	R
CCPC-TE	R (Thm. 4)	R
CCRPC-TP	R†	R
DCRPC-TP $=^*$ DCPC-TP	R* (Thm. 7)	R
CCPC-TP	R (Thm. 4)	R

Table 1: Complexity results for control by partition for veto in comparison with plurality

We have also identified and fixed a technical flaw in a very clever reduction due to Chen et al. [8]. Their reduction concerns the parameterized complexity of control by adding candidates to plurality elections, parameterized by the number of voters.

Regarding future work, a quite challenging interesting open question is to completely characterize the class of scoring protocols in terms of control complexity (i.e., to establish dichotomy results for the various control types), as has been done by Hemaspaandra et al. [24] for constructive control by adding voters, by Hemaspaandra and Hemaspaandra [20] for constructive coalitional weighted manipulation, and by Betzler and Dorn [6] and Baumeister and Rothe [4] for the possible winner problem (a generalization of coalitional unweighted manipulation due to Konczak and Lang [25]). Finally, it would also be interesting to study veto with respect to the refined models of control by partition introduced by Erdélyi et al. [13].

ACKNOWLEDGEMENTS

We thank the reviewers for many helpful suggestions. This work has been supported in part by DFG grant RO-1202/15-1.

[7] We omit the order symbol $\succ$, so the orders $R(i,j) \succ d \succ \cdots$ and $R'(i,j) \succ d \succ \cdots$ in this quote are written $R(i,j)\ d\ \cdots$ and $R'(i,j)\ d\ \cdots$ here.

REFERENCES

[1] Y. Bachrach, O. Lev, Y. Lewenberg, and Y. Zick, 'Misrepresentation in district voting', in *Proceedings of the 25th International Joint Conference on Artificial Intelligence*. AAAI Press/IJCAI, (July 2016). To appear.

[2] J. Bartholdi III, C. Tovey, and M. Trick, 'The computational difficulty of manipulating an election', *Social Choice and Welfare*, **6**(3), 227–241, (1989).

[3] J. Bartholdi III, C. Tovey, and M. Trick, 'How hard is it to control an election?', *Mathematical and Computer Modelling*, **16**(8/9), 27–40, (1992).

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

[5] D. Baumeister and J. Rothe, 'Preference aggregation by voting', in *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, ed., J. Rothe, chapter 4, 197–325, Springer-Verlag, (2015).

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

[7] N. Betzler and J. Uhlmann, 'Parameterized complexity of candidate control in elections and related digraph problems', *Theoretical Computer Science*, **410**(52), 5425–5442, (2009).

[8] J. Chen, P. Faliszewski, R. Niedermeier, and N. Talmon, 'Elections with few voters: Candidate control can be easy', in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pp. 2045–2051. AAAI Press, (January 2015).

[9] V. Conitzer, T. Sandholm, and J. Lang, 'When are elections with few candidates hard to manipulate?', *Journal of the ACM*, **54**(3), Article 14, (2007).

[10] R. Downey and M. Fellows, *Parameterized Complexity*, Springer-Verlag, 2nd edn., 2013.

[11] E. Elkind, P. Faliszewski, and A. Slinko, 'Cloning in elections: Finding the possible winners', *Journal of Artificial Intelligence Research*, **42**, 529–573, (2011).

[12] G. Erdélyi, M. Fellows, J. Rothe, and L. Schend, 'Control complexity in Bucklin and fallback voting: A theoretical analysis', *Journal of Computer and System Sciences*, **81**(4), 632–660, (2015).

[13] G. Erdélyi, E. Hemaspaandra, and L. Hemaspaandra, 'More natural models of electoral control by partition', in *Proceedings of the 4th International Conference on Algorithmic Decision Theory*, pp. 396–413. Springer-Verlag *Lecture Notes in Artificial Intelligence #9346*, (September 2015).

[14] G. Erdélyi, M. Nowak, and J. Rothe, 'Sincere-strategy preference-based approval voting fully resists constructive control and broadly resists destructive control', *Mathematical Logic Quarterly*, **55**(4), 425–443, (2009).

[15] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra, 'How hard is bribery in elections?', *Journal of Artificial Intelligence Research*, **35**, 485–532, (2009).

[16] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra, 'Multimode control attacks on elections', *Journal of Artificial Intelligence Research*, **40**, 305–351, (2011).

[17] P. Faliszewski, E. Hemaspaandra, L. Hemaspaandra, and J. Rothe, 'Llull and Copeland voting computationally resist bribery and constructive control', *Journal of Artificial Intelligence Research*, **35**, 275–341, (2009).

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

[19] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, 1979.

[20] E. Hemaspaandra and L. Hemaspaandra, 'Dichotomy for voting systems', *Journal of Computer and System Sciences*, **73**(1), 73–83, (2007).

[21] E. Hemaspaandra, L. Hemaspaandra, and C. Menton, 'Search versus decision for election manipulation problems', Technical Report arXiv:1202.6641 [cs.GT], Computing Research Repository, arXiv.org/corr/, (2012).

[22] E. Hemaspaandra, L. Hemaspaandra, and C. Menton, 'Search versus decision for election manipulation problems', in *Proceedings of the 30th Annual Symposium on Theoretical Aspects of Computer Science*, volume 20 of *LIPIcs*, pp. 377–388. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, (February 2013).

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

[24] E. Hemaspaandra, L. Hemaspaandra, and H. Schnoor, 'A control dichotomy for pure scoring rules', in *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, pp. 712–720. AAAI Press, (July 2014).

[25] K. Konczak and J. Lang, 'Voting procedures with incomplete preferences', in *Proceedings of the Multidisciplinary IJCAI-05 Workshop on Advances in Preference Handling*, pp. 124–129, (July/August 2005).

[26] A. Lin, *Solving Hard Problems in Election Systems*, Ph.D. dissertation, Rochester Institute of Technology, Rochester, NY, USA, March 2012.

[27] H. Liu and D. Zhu, 'Parameterized complexity of control problems in maximin election', *Information Processing Letters*, **110**(10), 383–388, (2010).

[28] A. Loreggia, N. Narodytska, F. Rossi, B. Venable, and T. Walsh, 'Controlling elections by replacing candidates or votes (extended abstract)', in *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems*, pp. 1737–1738. IFAAMAS, (May 2015).

[29] C. Menton, 'Normalized range voting broadly resists control', *Theory of Computing Systems*, **53**(4), 507–531, (2013).

[30] C. Menton and P. Singh, 'Control complexity of Schulze voting', in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pp. 286–292. AAAI Press/IJCAI, (August 2013).

[31] R. Niedermeier, *Invitation to Fixed-Parameter Algorithms*, Oxford University Press, 2006.

[32] D. Parkes and L. Xia, 'A complexity-of-strategic-behavior comparison between Schulze's rule and ranked pairs', in *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pp. 1429–1435. AAAI Press, (July 2012).

[33] C. Puppe and A. Tasnádi, 'Optimal redistricting under geographical constraints: Why "pack and crack" does not work', *Economics Letters*, **105**(1), 93–96, (2009).

[34] N. Russel, *Complexity of Control of Borda Count Elections*, Master's thesis, Rochester Institute of Technology, 2007.