Toward Completing the Picture of Control in Schulze and Ranked Pairs Elections

Cynthia Maushagen, David Niclaus, Paul Nüsken, Jörg Rothe, Tessa Seeger

Heinrich-Heine-Universität Düsseldorf, MNF, Institut für Informatik
{cynthia.maushagen, david.niclaus, paul.nuesken, rothe, tessa.seeger}@hhu.de

Abstract

Both the Schulze and the ranked pairs method are voting systems that satisfy many natural, desirable axioms. Many standard types of electoral control (where an external agent tries to change the outcome of an election by interfering with the election structure) have already been studied for them. However, for control by replacing candidates or voters and for multimode control that combines multiple standard attacks, many questions remain open. We solve a number of these open cases for Schulze and ranked pairs. In addition, we fix a flaw in the reduction of (Menton and Singh 2013, Thm. 2.2) showing that Schulze is resistant to constructive control by deleting candidates.

Introduction

Elections play a fundamental role in decision-making processes of modern societies and over time many voting systems have been established. The Schulze method (Schulze 2011, 2023) is a relatively new voting system and has gained unusual popularity over the past two decades due to its outstanding axiomatic properties. Although winner determination with the Schulze method is fairly complicated compared to most other voting systems, it can still be computed in polynomial time (Schulze 2011).

The ranked pairs method was specifically designed by Tideman (Tideman 1987) to satisfy the independence of clones criterion. In general, its axiomatic properties are as outstanding as Schulze's. However, that ranked pairs is barely widespread might be due to the fact that winner determination depends strongly on the handling of ties: When using "parallel universe tie-breaking" (Conitzer, Rognlie, and Xia 2009), the winner determination problem is NP-complete. It becomes tractable, however, when defining ranked pairs as a resolute rule by using a fixed tie-breaking method (Brill and Fischer 2012).

We study *electoral control* where a so-called *(election) chair* attempts to change the outcome of an election by changing its structure. Common examples are adding, deleting, partitioning (Bartholdi III, Tovey, and Trick 1992; Hemaspaandra, Hemaspaandra, and Rothe 2007), or replacing (Loreggia et al. 2015) candidates or voters. In addition, we also study multimode control (Faliszewski, Hemaspaandra, and Hemaspaandra 2011) where the chair can combine several attacks. For each corresponding decision problem, there is a *constructive* case (Bartholdi III, Tovey, and Trick 1992), where the goal is to make a favored candidate win, and a *destructive* case (Hemaspaandra, Hemaspaandra, and Rothe 2007), where the chair's aim is to prevent a despised candidate from winning.

Related Work Electoral control was introduced by Bartholdi et al. (Bartholdi III, Tovey, and Trick 1992) for the constructive cases and later on by Hemaspaandra et al. (Hemaspaandra, Hemaspaandra, and Rothe 2007) for the destructive cases. Control by replacing candidates or voters was introduced by Loreggia et al. (Loreggia et al. 2015), while Faliszewski et al. (Faliszewski, Hemaspaandra, and Hemaspaandra 2011) introduced and studied multimode control. Erdélyi et al. (Erdélyi et al. 2021) provide an extensive study and overview of various control problems, including replacing candidates or voters, and also multimode control. Other types of strategic influence on elections are *manipulation* and *bribery* (see, e.g., (Bartholdi III and Orlin 1991; Bartholdi III, Tovey, and Trick 1989; Conitzer and Sandholm 2006; Faliszewski, Hemaspaandra, and Hemaspaandra 2009; Faliszewski et al. 2009)). Control, bribery, and manipulation have been studied for a wide range of voting rules, as surveyed by Faliszwski and Rothe (Faliszewski and Rothe 2016) (bribery and control) and Conitzer and Walsh (Conitzer and Walsh 2016) (manipulation), see also (Baumeister and Rothe 2015).

For Schulze and ranked pairs, Parkes and Xia (Parkes and Xia 2012), Xia et al. (Xia et al. 2009), Menton and Singh (Menton and Singh 2013) and Gaspers et al. (Gaspers et al. 2013) studied constructive and destructive control by adding or deleting voters or candidates, bribery, and manipulation. Table 1 gives an overview of known results. Hemaspaandra et al. (Hemaspaandra, Lavaee, and Menton 2013) showed fixed-parameter tractability for bribing, controlling, and manipulating Schulze and ranked pairs elections with respect to the number of candidates and provided algorithms with uniform polynomial running time that are independent of the number of candidates. Menton and Singh (Menton and Singh 2013) also provided results on control by partition and runoff partition of candidates and partition of voters for Schulze and further showed some results for all Condorcet-consistent voting rules.

Table 1: Overview of results on the complexity of standard control (AC, DC, AV, DV), bribery (B), and manipulation (M) in Schulze and ranked pairs elections. R means resistance, V vulnerability, S susceptibility (as defined at the end of the Prelimiaries); our results are in **blue**. Results marked by $\spadesuit$ are due to Parkes and Xia (Parkes and Xia 2012), by $\blacklozenge$ first claimed to be V, then withdrawn by Menton and Singh (Menton and Singh 2012) and for DC now re-established in Theorem 5, and by $\clubsuit$ due to Xia et al. (Xia et al. 2009). $\bigstar$ marks a result by Parkes and Xia (Parkes and Xia 2012), extended by Gaspers et al. (Gaspers et al. 2013). The original proof of Menton and Singh (Menton and Singh 2013) for the result marked by $\diamondsuit$ is disproven and fixed in the proof of Theorem 1.

		AC	DC	AV	DV	B	M
Schulze	Constructive	R$^{\spadesuit}$	**R**$^{\diamondsuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	V$^{\bigstar}$
	Destructive	S$^{\blacklozenge}$	**V**$^{\blacklozenge}$	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	V$^{\spadesuit}$
Ranked pairs	Constructive	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\clubsuit}$
	Destructive	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\spadesuit}$	R$^{\clubsuit}$

Preliminaries

An *election* is a pair (C, V), where $C = \{c_1, \dots, c_m\}$ is a set of *candidates* and V is a list of n votes in form of *preferences* over all candidates in C. Preferences can be expressed either as approval sets or as preference orders. In this paper, we focus on the latter, where voters rank the candidates in descending order from most to least preferred. Formally, a preference order is a strict linear order over C. We write $c \succ_{v_i} d$ to express that a voter $v_i \in V$ prefers candidate c over d. When it is clear from the context, we omit $\succ_{v_i}$ and simply write $c\,d$. For a set of candidates $A \subseteq C$, we refer to the lexicographic order of those candidates by simply writing $\overrightarrow{A}$, and $\overleftarrow{A}$ for the reverse. If only A occurs in a vote, the candidates from A are ranked in lexicographic order. For example, $c\,A$ is the same as $c\,\overrightarrow{A}$ and denotes the vote that ranks c first and then all candidates from A follow in lexicographic order. For a set of candidates C and two candidates $c, d \in C$, we will write $W(c, d)$ for the two votes $c\,d\,\overrightarrow{C \setminus \{c, d\}}$ and $\overleftarrow{C \setminus \{c, d\}}\,c\,d$. For a given election (C, V), let $N_V(c, d)$ be the number of votes, in which candidate c is ranked above candidate d. Note that we omit the list of votes if it is clear from the context. A voting rule $r : \{(C, V) \mid (C, V) \text{ is an election}\} \to 2^C$ determines the set of winners of an election (C, V). We focus on *Schulze* and *ranked pairs*. Both are *Condorcet voting systems*, meaning they always choose the candidate as the winner who wins in the pairwise comparison against any other candidate, if there is one. Such a candidate is called a *Condorcet winner*.

Schulze: For an election (C, V), we first construct the *weighted majority graph (WMG)*. A WMG is a weighted directed graph $G = (\hat{V}, E, w)$, where $\hat{V} = C$ and there is an edge from candidate c to d with weight $D_V(c, d) = N_V(c, d) - N_V(d, c)$ if $N_V(c, d) > N_V(d, c)$. Let the *strength of a path* p ($\text{str}(p)$) be the weight of the weakest edge in the path p. For each pair of candidates $c, d \in C$, let the *strength of the strongest path* be $P(c, d) = \max\{\text{str}(p) \mid p \text{ is a path from } c \text{ to } d\}$ if there exists a path from c to d; otherwise, set $P(c, d) = -\infty$. A candidate $c \in C$ is a *Schulze winner of* (C, V) if $P(c, d) \geq P(d, c)$ for each $d \in C \setminus \{c\}$.

Example 1. *Consider an election (C, V) with the candidate set $C = \{a, b, c, d\}$ and the following votes:*

$$\begin{aligned} 4 \times v_1 &: \ a\ c\ b\ d, \\ 2 \times v_2 &: \ d\ a\ c\ b, \\ 3 \times v_3 &: \ d\ c\ a\ b, \\ 2 \times v_4 &: \ b\ d\ a\ c. \end{aligned}$$

First, we determine $N_V(x, y)$ for each pair of candidates $x, y \in C$ to build the WMG:

$$\begin{array}{lll} N_V(a,b) = 9, & N_V(a,c) = 8, & N_V(a,d) = 4, \\ N_V(b,a) = 2, & N_V(b,c) = 2, & N_V(b,d) = 6, \\ N_V(c,a) = 3, & N_V(c,b) = 9, & N_V(c,d) = 4, \\ N_V(d,a) = 7, & N_V(d,b) = 5, & N_V(d,c) = 7. \end{array}$$

The WMG is shown in Figure 1. The strengths of the strongest paths for each pair of candidates $x, y \in C$ are given in Table 2. Since $P(d, x) \geq P(x, d)$ for each $x \in C \setminus \{d\}$ and this does not hold for any other candidate, d is the unique Schulze winner of the election.

Ranked pairs: For an election (C, V), we first calculate $D_V(c, d)$ for all pairs of candidates $c, d \in C, c \neq d$, and then order the pairs from highest to lowest weights, i.e., we order the values of $D_V(c, d)$. We consider each pair of candidates one by one in the order determined by the weights $D_V(c, d)$. Note that following Parkes and Xia (Parkes and Xia 2012), we break ties according to a fixed tie-breaking rule. Now, in each step, we consider the highest ranking pair c, d of the weight order, which has not yet been considered. We fix the order $c \succ d$ as long as the relation $c \succ d$ does not violate transitivity (which would contradict the previously fixed order, in which case this pair is disregarded). When all pairs have been considered, the *ranked pairs winner of the election* (C, V) (subject to the fixed tie-breaking) is the highest ranking candidate of the resulting order of candidates.

Since we do not need a complete ranking of the candidates, we use a slightly simplified but equivalent definition of ranked pairs introduced by Berker et al. (Berker et al. 2022), which only returns the winner of the election and works by constructing an acyclic graph. After ordering the positive majorities $D_V(c, d)$ in decreasing order, in each step, we consider the top pair (c, d) of this weight order, which has not yet been considered. Again, ties are broken by a fixed tie-breaking method. We add an edge (c, d) to a directed graph G, unless inserting this edge would create a cycle, in which case the pair (i.e., the edge) is disregarded. When all pairs have been considered, the *ranked pairs winner of* (C, V) (subject to the fixed tie-breaking) is the candidate corresponding to the source of G.

Example 2. *Consider the election described in Example 1. By using lexicographical tie-breaking, we get the weight order shown in Table 3. We now start by considering the first*

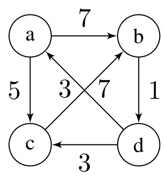

Figure 1: Weighted majority graph in Example 1.

Table 2: Strengths of the strongest paths in Example 1.

$P(x,y)$	a	b	c	d
a	–	7	5	1
b	1	–	1	1
c	1	7	–	1
d	3	3	3	–

Table 3: Order of weights in Example 2.

	pair (c_i, c_j)	$D_V(c_i, c_j)$
1.	(a, b)	7
2.	(c, b)	7
3.	(a, c)	5
4.	(d, a)	3
5.	(d, c)	3
6.	(b, d)	1

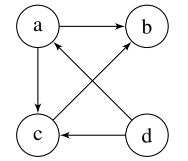

Figure 2: Directed graph $\boldsymbol{G}$ in Example 2.

pair (a, b). The corresponding edge (a, b) can of course be inserted into the directed graph G. In the next four steps, one after another we consider the pairs (or, edges) (c, b), (a, c), (d, a), and (d, c), which can all be added to G since none of them creates a cycle. Only the last edge (b, d) cannot be inserted since we would then get a cycle among b, c, and d. The directed graph G is depicted in Figure 2. Being the source of the graph, d is the ranked pairs winner of the election.

We will study various types of electoral control, starting with constructive control by deleting candidates, which has been defined by Bartholdi et al. (Bartholdi III, Tovey, and Trick 1992) for an election system $\mathcal{E}$ as follows:

$\mathcal{E}$-CONSTRUCTIVE CONTROL BY DELETING CANDIDATES	
Given:	An election (C, V), a distinguished candidate $p \in C$, and $\ell \in \mathbb{N}$.
Question:	Is it possible to make p the unique winner of the $\mathcal{E}$ election resulting from (C, V) by deleting at most ℓ candidates?

In the setting of replacing candidates or voters (Loreggia et al. 2015; Erdélyi et al. 2021), the chair must not alter the size of the election and instead must add a candidate or voter for each one she deletes.

$\mathcal{E}$-CONSTRUCTIVE CONTROL BY REPLACING CANDIDATES	
Given:	An election (C, V), a set of unregistered candidates D, $C \cap D = \emptyset$, a distinguished candidate $p \in C$, and $\ell \in \mathbb{N}$.
Question:	Is it possible to make p the unique winner of the $\mathcal{E}$ election resulting from (C, V) by replacing at most ℓ candidates $C' \subseteq C$ with candidates $D' \subseteq D$, where $\lvert C' \rvert = \lvert D' \rvert$?

$\mathcal{E}$-CONSTRUCTIVE CONTROL BY REPLACING VOTERS ($\mathcal{E}$-CCRV) is defined analogously by asking whether it is possible to make a preferred candidate p the unique winner of the $\mathcal{E}$ election resulting from (C, V) by replacing at most ℓ votes $V' \subseteq V$ with votes $U' \subseteq U$ such that $\lvert V' \rvert = \lvert U' \rvert$, where U is a list of as of yet unregistered votes.

In these control scenarios, a chair's goal is to make a preferred candidate the unique winner. A chair may also be interested in preventing a candidate from winning. This setting is known as destructive control (Hemaspaandra, Hemaspaandra, and Rothe 2007). Instead of asking whether a candidate can be made a winner, we ask whether the sole victory of a candidate can be prevented. We abbreviate control problems by the first letter of each word, e.g. $\mathcal{E}$-DCDC for $\mathcal{E}$-DESTRUCTIVE CONTROL BY DELETING CANDIDATES. Aside from the *unique-winner model*, which is used for the given definitions of the above control problems, in the *nonunique-winner model* we ask whether a preferred candidate can be made a winner (possibly among others) in the constructive case, and whether a despised candidate can be prevented from winning altogether in the destructive case. Note that when interpreting a voting rule as *resolute* (i.e., to always yield exactly one winner due to tie-breaking), the unique-winner and nonunique-winner models are the same.

In addition to the above control problems, we also study variations of multimode control (Faliszewski, Hemaspaandra, and Hemaspaandra 2011), where several of the standard control actions and bribery can be combined into one attack. For an election system $\mathcal{E}$, the problem is defined as follows:

$\mathcal{E}$-CONSTRUCTIVE CONTROL BY AC+DC+AV+DV+B	
Given:	Two disjoint sets of candidates, C and D, two disjoint lists of votes over $C \cup D$, V and U, a distinguished candidate $p \in C$, and ℓ_{AC}, ℓ_{DC}, ℓ_{AV}, ℓ_{DV}, $\ell_B \in \mathbb{N}$.
Question:	Is it possible to find two sets, $C' \subseteq C \setminus \{p\}$ and $D' \subseteq D$, and two sublists of votes, $V' \subseteq V$ and $U' \subseteq U$, such that p is the unique winner of the $\mathcal{E}$ election that results from $((C \setminus C') \cup D', (V \setminus V') \cup U')$ by changing (i.e., bribing) at most ℓ_B votes in $(V \setminus V') \cup U')$, and $\lvert D' \rvert \leq \ell_{AC}$, $\lvert C' \rvert \leq \ell_{DC}$, $\lvert U' \rvert \leq \ell_{AV}$, and $\lvert V' \rvert \leq \ell_{DV}$?

We abbreviate multimode control problems in the obvious way; e.g., we use the shorthand $\mathcal{E}$-CCAC+DC+AV+DV+B for the above problem. Faliszewski et al. (Faliszewski, Hemaspaandra, and Hemaspaandra 2011) define a method to classify all $2^5 - 1$ variants of multimode control for an election system, called classification rule A. Part of this classification rule is (Faliszewski, Hemaspaandra, and Hemaspaandra 2011, Theorem 4.7), which states that for any combination of attack prongs, where the voting rule is resistant to at least one, it is also resistant to that combination. Using this and the known results for adding and deleting candidates or voters and for bribery (see Table 1), it immediately follows that, except for Schulze-DCAC+DC, Schulze is resistant to any multimode attack. Since ranked pairs is resistant to all single-pronged attacks, it is clear that ranked pairs also resists all combinations of multimode control.

In the exact setting, $\mathcal{E}$-EXACT CONSTRUCTIVE CONTROL BY AC+DC+AV+DV+B, it must hold that $\lvert D' \rvert =$

ℓ_{AC}, $|C'| = \ell_{DC}$, $|U'| = \ell_{AV}$, $|V'| = \ell_{DV}$, and exactly ℓ_B voters in $(V \setminus V') \cup U'$ were bribed. The destructive variants are defined analogously by asking whether it is possible to make p not a (unique) winner, and we again use the obvious shorthands. Sometimes, we exclude certain control/bribery actions from multimode control, considering, e.g., only candidate control ($\mathcal{E}$-EDCAC+DC) and then omit the unneeded input parameters U, ℓ_{AV}, ℓ_{DV}, and ℓ_B. Note that we do not allow candidates in D or voters in U to be deleted.

We say a voting system is *immune to a type of control* if it is impossible to change the outcome of an election by that control type; otherwise, this voting system is *susceptible to that type of control*. When we have susceptibility to some control type, we say this voting system is *resistant to this control type* if the corresponding control problem is NP-hard, and it is *vulnerable to it* if the corresponding control problem is solvable in polynomial time.

Constructive Control by Deleting Candidates

We now fix a flaw in the proof of the following result.

Theorem 1. *Schulze voting is resistant to constructive control by deleting candidates, i.e., Schulze-*CCDC *is* NP*-complete in the nonunique-winner model.*

Proof. The proof of this result, due to Menton and Singh (Menton and Singh 2013), shows a clever reduction from 3SAT, but it is technically flawed. We briefly present their reduction from the proof of (Menton and Singh 2013, Thm. 2.2) and give a counterexample.

In the 3-SATISFIABILITY problem (3SAT), we are given a set X of variables and a set $Cl = \{Cl_1, \dots, Cl_k\}$ of clauses over X, each having exactly three literals, and we ask whether there is a satisfying assignment for φ, where φ is the conjunction of all clauses $Cl_i \in Cl$. Given a 3SAT instance (X, Cl), Menton and Singh (Menton and Singh 2013) construct a Schulze-CCDC instance $((C, V'), p, k)$ as follows. The set of candidates C contains

- $k + 1$ *clause candidates* $c_i^1, \dots, c_i^{k+1}$ for each clause $Cl_i \in Cl$,
- three *literal candidates* x_i^1, x_i^2, x_i^3 for each clause Cl_i, where x_i^j is the *j*th literal in clause Cl_i,
- $k+1$ *negation candidates* $n_{i,j,m,n}^1, \dots, n_{i,j,m,n}^{k+1}$ for each pair of literals x_i^j, x_m^n, where one is the negation of the other, and
- the distinguished candidate p and an additional candidate a.

Let $C_i = \{c_i^1, \dots, c_i^{k+1}\}$ be the set of all clause candidates for clause $Cl_i \in Cl$ and let $K = \bigcup_{i=1}^{k+1} C_i$ be the set of all clause candidates. Let $L_i = \{x_i^1, x_i^2, x_i^3\}$ be the set of literal candidates for the clause Cl_i and let $L = \bigcup_{i=1}^{k+1} L_i$ be the set of all literal candidates. Let $N_{ijmn} = \{n_{ijmn}^1, \dots, n_{ijmn}^{k+1}\}$ be the set of negation candidates for the literals x_i^j, x_m^n that are a negation of each other, and let N be the set of all such negation candidates. For a positive integer z, we write $[z] = \{1, \dots, z\}$ as a shorthand. Menton and Singh (Menton and Singh 2013) define the following list of votes V' (which we will change later to fix the proof):

#	preferences	for each
1	$W(c_i^j, x_i^1)$	$i \in [k]$, $j \in [k+1]$
1	$W(x_i^1, x_i^2)$	$i \in [k]$
1	$W(x_i^2, x_i^3)$	$i \in [k]$
1	$W(x_i^3, p)$	$i \in [k]$
1	$W(n_{ijmn}^l, p)$	$n_{ijmn}^l \in N$
1	$W(a, x_i^j)$	$x_i^j \in L$
1	$W(x_i^j, n_{ijmn}^l)$	$l \in [k+1]$; x_m^n is the negation of x_i^j
1	$W(p, a)$	

The deletion limit is k, the number of clauses. Menton and Singh (Menton and Singh 2013) argue that p can be made a Schulze winner by deleting at most k candidates from C if and only if there exists a truth assignment that makes the given 3SAT instance true. However, the following example poses a counterexample to the correctness of their proof.

Example 3. *We start with a yes-instance of* 3SAT, *which will be mapped to a no-instance of Schulze-*CCDC *by their reduction. Let (X, Cl) be our given* 3SAT *instance, with $X = \{x_1, x_2, x_3\}$ and $Cl = \{(x_1 \vee x_2 \vee \neg x_3), (\neg x_1 \vee x_2 \vee x_3)\}$, i.e., we consider the CNF formula*

$$\varphi = (x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee x_3)\,.$$

*According to the proof of Menton and Singh (Menton and Singh 2013, Thm. 2.2), we construct an instance $((C, V'), p, \ell)$ of Schulze-*CCDC *as follows. The deletion limit is $k = 2$, the number of clauses $|Cl|$. The set of candidates, $C = K \cup L \cup N \cup \{p, a\}$, consists of*

- *the clause candidates c_1^1, c_1^2, c_1^3, c_2^1, c_2^2, and c_2^3,*
- *the literal candidates x_1^1, x_1^2, x_1^3, x_2^1, x_2^2, and x_2^3,*
- *the negation candidates $n_{1,1,2,1}^1$, $n_{1,1,2,1}^2$, $n_{1,1,2,1}^3$, $n_{1,3,2,3}^1$, $n_{1,3,2,3}^2$, and $n_{1,3,2,3}^3$ (abbreviated by n_1^1, n_1^2, n_1^3, n_2^1, n_2^2, and n_2^3),*
- *the distinguished candidate p and the additional candidate a.*

The preferences are represented by the WMG in Figure 3 using McGarvey's trick (McGarvey 1953). Each edge in this graph has a weight of two.

Before control, every clause candidate c_i^j, with $1 \leq i \leq 2$ and $1 \leq j \leq 3$, ties the other clause candidates and has a path to every other candidate while no candidate has a path to this clause candidate.[1] *Each remaining candidate, including the distinguished candidate p, ties every other candidate except for the six clause candidates, to whom they lose. Thus the six clause candidates c_1^1, c_1^2, c_1^3, c_2^1, c_2^2, and c_2^3 are the Schulze winners of the election. Now we want to delete at most two candidates to make p a winner of the election. Consider the assignment $x_1 =$* TRUE, *which satisfies the first clause, and $x_2 =$* TRUE, *which satisfies the second clause of*

[1]Note that all paths have a strength of two, as there are no other edge weights in the weighted majority graph.

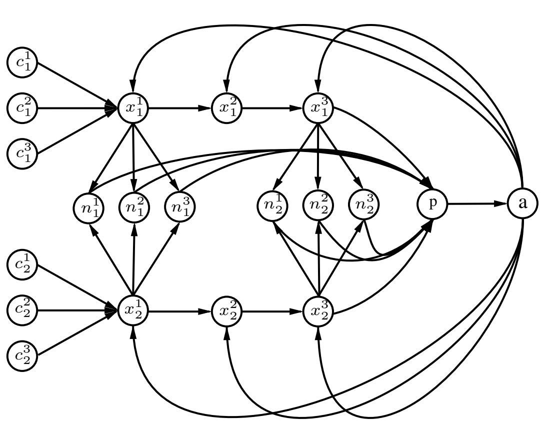

Figure 3: Weighted majority graph corresponding to the election $(\boldsymbol{C}, \boldsymbol{V'})$ constructed from a 3SAT instance in Example 3. All edges have a weight of two.

the CNF formula φ, i.e., we have a yes-instance of 3SAT.[2] *To ensure that p is a winner of the election, it is necessary to destroy each path from a clause candidate to p. Menton and Singh (Menton and Singh 2013) argue that in order to do so, it is sufficient to* "delete one literal candidate for each clause, selecting a literal that is satisfied by the satisfying assignment for Cl." *However, if we delete the literal candidates x_1^1 and x_2^2, there is still a path from the candidates c_2^1, c_2^2, and c_2^3, traversing x_2^1 and n_1^i with $1 \leq i \leq 3$, to the distinguished candidate p. Therefore, $P(c_2^i, p) > P(p, c_2^i)$ for $1 \leq i \leq 3$ and thus p is still not a Schulze winner of the election. We now argue that p cannot be made a Schulze winner of the election by deleting any other possible choice of $k = 2$ candidates, so we indeed have a no-instance of Schulze-*CCDC. *First, there are $k+1 = 3$ candidates in each group of clause candidates C_i and negation candidates N_{ijmn}, and since all of them have the same incoming and outgoing edges in the WMG, deleting only two of them cannot make p a Schulze winner of the election. Further, deleting candidate a would result in p losing to all literal candidates in L. Therefore, the only way to guarantee p's victory is to delete two of the literal candidates x_1^j, x_2^j for $j \in [k+1]$. It is easy to see that at least one literal candidate for each clause must be deleted to break the paths between all clause candidates and p. It thus suffices to consider only those pairs of literal candidates where $i = 1$ for one and $i = 2$ for the other. The following table lists those candidates to whom p loses when the pair in the corresponding column is deleted, thus preventing p from becoming a Schulze winner of the election:*

deleted pair	x_1^1 x_2^1	x_1^1 x_2^2	x_1^1 x_2^3	x_1^2 x_2^1	x_1^2 x_2^2	x_1^2 x_2^3	x_1^3 x_2^1	x_1^3 x_2^2	x_1^3 x_2^3
wins against p	n_1^j	c_2^j	c_2^j	c_1^j	c_i^j	c_i^j	c_1^j	c_i^j	c_i^j, n_2^j

Note that $j \in [k+1]$ and $i \in [2]$. This shows that a yes-instance of 3SAT *has been mapped to a no-instance of Schulze-*CCDC *by the reduction in the proof of Menton and Singh (Menton and Singh 2013, Thm. 2.2).*

[2]The truth assignment of x_3 is irrelevant, as both clauses and thus the formula φ are already true due to the truth assignment of x_1 and x_2.

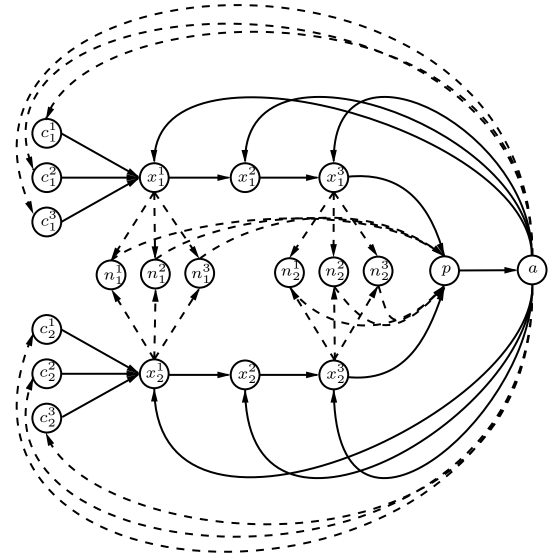

Figure 4: WMG corresponding to the election $(\boldsymbol{C}, \boldsymbol{V})$ constructed from the given 3SAT instance from our counterexample with the new reduction. Dashed edges have a weight of two and drawn edges have a weight of four.

Their reduction is quite clever, but unfortunately wrong, as shown by the counterexample. However, by modifying it appropriately, we can ensure that p can indeed be made a Schulze winner of the election by deleting at most k candidates if and only if (X, Cl) is a yes-instance of 3SAT. For our modifications, it is only necessary to change the list of votes. The list of votes V over the same candidates C is now the following:

#	preferences	for each
2	$W(c_i^j, x_i^1)$	$i \in [k], j \in [k+1]$
2	$W(x_i^1, x_i^2)$	$i \in [k]$
2	$W(x_i^2, x_i^3)$	$i \in [k]$
2	$W(x_i^3, p)$	$i \in [k]$
2	$W(a, x)$	$x \in L$
2	$W(p, a)$	
1	$W(n, p)$	$n \in N$
1	$W(a, c)$	$c \in K$
1	$W(x_i^j, n_{ijmn}^l)$	$l \in [k+1]$; x_m^n is the negation of x_i^j

The graph in Figure 4 shows the weighted majority graph for the 3SAT instance from our counterexample adapted to the new reduction. We claim that (X, Cl) is a yes-instance of 3SAT if and only if $((C, V), p, k)$ is a yes-instance of Schulze-CCDC in the nonunique-winner model.

From left to right, let (X, Cl) be a yes-instance of 3SAT. Since we have a yes-instance of 3SAT, we have a truth assignment that makes at least one literal in each clause $Cl_i \in Cl$ true. We claim that p can be made a Schulze winner by deleting one literal candidate corresponding to some true literal for each clause. The only path with weight four from a clause candidate $c_i^j \in K$ to p is through the literal candidates: from x_i^1 via x_i^2 to x_i^3. Since we deleted one literal candidate for each clause, there no longer exists a weight-4 path from a clause candidate to p and $P(p, c) = 2 \geq P(c, p)$ for $c \in K$. For each $c \in L \cup \{a\}$, we have $P(p, c) = 4 = P(c, p)$. Since we deleted only literal candidates where the corresponding literal was assigned to be true, we never have the case that we deleted two literal candidates x_i^j, x_m^n, which negate each other. Thus we still have a weight-2 path from p to each negation candidate and $P(p, c) = 2 = P(c, p)$ for each $c \in N$. It follows that $((C, V), p, k)$ is a yes-instance of Schulze-CCDC in the nonunique-winner model.

From right to left, let (X, Cl) be a no-instance of 3SAT. Thus, for each assignment of the literals, there exists a clause which is false. To ensure that p is a winner of the election, it is necessary that $P(p, c) \geq P(c, p)$ for each $c \in C \setminus \{p\}$. Since $P(p, c) = 2 < 4 = P(c, p)$ for each $c \in K$, we have to destroy each path of weight greater than two from the clause candidates to p, in particular the path through the literal candidates $x_i^j \in C_i$, $j \in \{1, 2, 3\}$. Due to the deletion limit $k = |Cl|$, it is necessary to delete one literal candidate for each clause. Consider any subset of literals of size k such that for each clause one literal is contained in the set. It follows that this set contains at least two literals, x_i^j and x_m^n, that negate each other, for otherwise, the formula would be satisfiable and we would have a yes-instance of 3SAT. By deleting the corresponding two literal candidates, we no longer have a path from p to their negation candidates $n \in N_{ijmn}$. It follows that $P(p, n) < P(n, p)$ and it is impossible to make p a Schulze winner of the election by deleting at most k candidates. Therefore, $((C, V), p, k)$ is a no-instance of Schulze-CCDC in the nonunique-winner model.

Finally, it is easy to see that Schulze-CCDC is in NP. □

Exact Multimode and Control by Replacing

Now we turn to exact multimode control and control by replacing candidates or voters. Lorregia et al. (Loreggia et al. 2015) showed that any voting rule that is resistant to constructive control by deleting candidates and satisfies *IBC* is also resistant to constructive control by replacing candidates.[3] We extend their result to also apply to $\mathcal{E}$-ECCAC+DC and $\mathcal{E}$-ECCRC.

Lemma 1. *Let $\mathcal{E}$ be a voting rule that satisfies IBC. If $\mathcal{E}$-*CCDC *is* NP*-hard, then so are $\mathcal{E}$-*ECCAC+DC *and $\mathcal{E}$-*ECCRC.

Proof. We reduce $\mathcal{E}$-CCDC to $\mathcal{E}$-ECCAC+DC and $\mathcal{E}$-ECCRC. Let (C, V, p, k) be an instance of $\mathcal{E}$-CCDC. Define $C' = C \cup X$ with $X = \{x_1, \dots, x_k\}$, $D = \{d_1, \dots, d_{\ell_{AC}}\}$, $\ell_{DC} = \ell_{RC} = k$, and set $\ell_{AC} \in \mathbb{N}$ arbitrarily. Let $V' = v\ X\ D$ for every $v \in V$, i.e., add all candidates from X at the bottom of every vote and then add all candidates from D at the bottom of those votes. Construct an instance $(C', D, V', p, \ell_{AC}, \ell_{DC})$ of $\mathcal{E}$-ECCAC+DC and an $\mathcal{E}$-ECCRC instance $((C', V'), p, \ell_{RC})$. Since candidates from D are bottom-ranked in all votes, they have no influence on the winner determination. Thus the outcome of the election can only be changed by the deletion attack. In a $\mathcal{E}$-CCDC yes-instance, a set of candidates whose removal guarantees p to win exists, which can be padded by candidates from X to allow for successful exact multimode control. A $\mathcal{E}$-CCDC no-instance cannot be recovered to allow for candidate p's victory in the multimode setting. □

Lemma 2. *Let $\mathcal{E}$ be a voting rule that satisfies IBC. If $\mathcal{E}$-*DCDC *is* NP*-hard, then so are $\mathcal{E}$-*EDCAC+DC*, $\mathcal{E}$-*DCRC*, and $\mathcal{E}$-*EDCRC.

Proof. The same construction and proof idea used in the proof of Lemma 1 works here as well. □

Lemma 3. *Schulze and ranked pairs are insensitive to bottom-ranked candidates.*

Proof. Let (C, V) be an election and x a new candidate. Let $(C \cup \{x\}, V^x)$ be the election where candidate x is added to every vote in V as the least preferred option. Clearly, we have $N_{V^x}(x, c) = 0$ and thus $D_{V^x}(x, c) = -|V^x|$ for all $c \in C \setminus \{x\}$. For Schulze, it follows that in the WMG candidate x has an incoming edge with weight $|V^x|$ for every $c \in C$ and the outdegree of x is 0. Thus candidate x cannot win and can also not be part of any path between any other candidates $c, c' \in C$. For ranked pairs, it follows that the pair (c, x) will be in the top ranking for each $c \in C$. These pairs will be fixed first and the winner of election $(C \cup \{x\}, V^x)$ is the same as of election (C, V). □

We now obtain the following three results.

Corollary 1. *Schulze-*CCRC *and Ranked-Pairs-*CCRC *are* NP*-complete.*

Proof. Schulze and ranked pairs are IBC (see Lemma 3), from Theorem 1 we know that Schulze-CCDC is NP-hard, and Parkes and Xia (Parkes and Xia 2012) showed NP-hardness of Ranked-Pairs-CCDC. Hence, by the result of Lorregia et al. (Loreggia et al. 2015), Schulze-CCRC and Ranked-Pairs-CCRC are also NP-hard. It is easy to see that Schulze-CCRC and Ranked-Pairs-CCRC are in NP and thus, NP-complete.[4] □

Corollary 2. *Schulze-*ECCAC+DC*, Schulze-*ECCRC*, Ranked-Pairs-*ECCAC+DC*, and Ranked-Pairs-*ECCRC *are* NP*-complete.*

Proof. By Lemma 1, we can extend the proof of Corollary 1 to Schulze-ECCAC+DC, Schulze-ECCRC, Ranked-Pairs-ECCAC+DC, and Ranked-Pairs-ECCRC. □

[3] IBC refers to *insensitivity to bottom-ranked candidates* (Lang, Maudet, and Polukarov 2013): A voting rule $\mathcal{E}$ is said to be *IBC* if, given an election (C, V) and a new candidate x, the election (C, V) and $(C \cup \{x\}, V^x)$, where V^x is the list of votes obtained by adding x as the least preferred alternative to each vote in V, have the same winners under $\mathcal{E}$.

[4] As noted earlier, we use a fixed tie-breaking scheme to ensure tractability.

Corollary 3. *Ranked-Pairs-*EDCAC+DC*, Ranked-Pairs-*EDCRC*, and Ranked-Pairs-*DCRC *are* NP*-complete.*

Proof. By Lemma 3, ranked pairs is IBC and Parkes and Xia (Parkes and Xia 2012) showed NP-hardness for Ranked-Pairs-DCDC. By the result of Lorregia et al. (Loreggia et al. 2015), Ranked-Pairs-DCRC is also NP-hard. Since Ranked-Pairs-DCRC is in NP, it is NP-complete. By Lemma 2, this also applies to Ranked-Pairs-EDCAC+DC and Ranked-Pairs-ECCRC. □

We now show resistance to exact multimode control and control by replacing voters by adapting the reductions for Schulze-CCAV (Menton and Singh 2013).

Theorem 2. *In both the unique-winner and the nonunique-winner model, Schulze-*ECCAV+DV*, Schulze-*CCRV*, Schulze-*EDCAV+DV*, and Schulze-*DCRV *are* NP*-complete.*

Proof sketch. To show NP-hardness, we reduce from the NP-complete problem RESTRICTED EXACT COVER BY 3-SETS (RX3C) (Gonzalez 1985): Given a set $B = \{b_1, \ldots, b_{3s}\}$ with $s \geq 1$ and a list $\mathcal{S} = \{S_1, \ldots, S_{3s}\}$, where $S_i = \{b_{i,1}, b_{i,2}, b_{i,3}\}$ and $S_i \subseteq B$ for all $S_i \in \mathcal{S}$ and each b_j is contained in exactly three sets $S_i \in \mathcal{S}$, does there exist an exact cover, i.e., a sublist $\mathcal{S}' \subseteq \mathcal{S}$ such that each $b_i \in B$ occurs in exactly one $S_i \in \mathcal{S}'$?

Let $(B, \mathcal{S})$ be an RX3C instance, where $B = \{b_1, \ldots, b_{3s}\}$, $\mathcal{S} = \{S_1, \ldots, S_n\}$ and for each $S_i \in \mathcal{S}$ we have $S_i = \{b_{i,1}, b_{i,2}, b_{i,3}\}$. In addition, each b_j is contained in exactly three sets $S_i \in \mathcal{S}$, giving us $|B| = |\mathcal{S}| = 3s$. We present the reduction in the nonunique-winner model; the reduction can be adapted for the unique-winner model. Let $\ell_{AV} = \ell_{DV} = s$ ($\ell_{RV} = s$) for the Schulze-ECCAV+DV and Schulze-EDCAV+DV (Schulze-CCRV and Schulze-DCRV) instances we construct. Further, let $\mathcal{L} \gg s$ be a constant much greater than s. From $(B, \mathcal{S})$ we construct an election (C, V) as follows. Let the candidate set be $C = B \cup \{p, w\}$. The list of votes contains ℓ_{DV} votes of the form $w\, B\, p$ and the remaining votes are constructed such that

- p beats w by $2\mathcal{L}$ votes,
- each b_i beats p by $2\mathcal{L} + 2\ell_{DV} + 2\ell_{AV} - 2$ votes,
- w beats each b_i by votes $\gg 2\mathcal{L}$ and
- all other pairwise differences are 0 or at least smaller than $\mathcal{L}$.

The WMG of the resulting election is shown in Figure 5. Note that candidate w is the unique winner of the election. The list of additional votes U contains one vote $S_i\, p\, (B \setminus S_i)\, w$ for each $S_i \in \mathcal{S}$. Let p be the distinguished candidate for the constructive case and w be the despised candidate for the destructive case. We claim that p can be made a Schulze winner (and w can be prevented from being the unique Schulze winner) by adding exactly ℓ_{AV} voters and deleting exactly ℓ_{DV} voters if and only if $(B, \mathcal{S})$ is a yes-instance of RX3C. □

By adapting the construction in the proof of Theorem 2, we obtain the same results for ranked pairs.

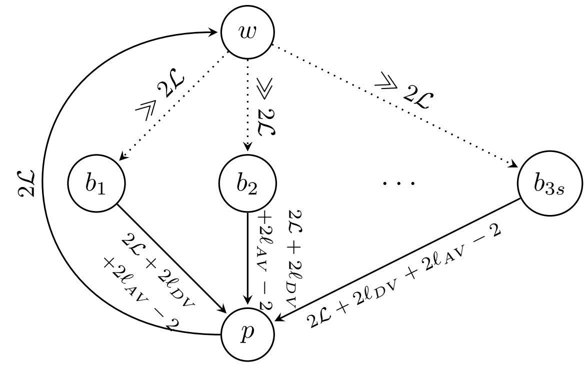

Figure 5: The WMG of the election (C, V) from the proof of Theorem 2.

Theorem 3. *In both the unique-winner and the nonunique-winner model, Ranked-Pairs-*ECCAV+DV*, Ranked-Pairs-*CCRV*, Ranked-Pairs-*EDCAV+DV*, and Ranked-Pairs-*DCRV *are* NP*-complete.*

Proof sketch. We only need to adjust the voter list as follows. Instead of having the pairwise differences between each pair of candidates $b_i, b_j \in B$ be zero, we have each b_i win by $\gg 2\mathcal{L}$ against all $b_j \in B$, where $i < j$, i.e.,

$$D(b_i, b_j) \gg 2\mathcal{L} \qquad \forall i < j,\ 1 \leq i, j \leq 3s.$$

□

Destructive Control by Deleting Candidates

In this section, we consider destructive control by deleting candidates. We examine a peculiarity of standard destructive control by deleting candidates for Schulze elections, which allows us to reduce the candidates we need to consider for deletion when solving the control problem. Using this approach, we are able to re-establish the result that DCDC is polynomial-time solvable for Schulze elections, which Menton and Singh claimed in an early version (v1) of their arXiv preprint (Menton and Singh 2012). However, Menton and Singh removed this result (and the corresponding result for Schulze-DCAC) from all subsequent versions of the arXiv preprint and from their IJCAI 2013 paper (Menton and Singh 2013), stating these two control problems as open.

Theorem 4. *If destructive control by deleting candidates is possible for a given Schulze election, then there exists some candidate $c \in C$ who can beat the despised candidate w by only deleting candidates who directly beat c (i.e., are in-neighbors of c in the WMG).*

Proof. Let $c \in C$ be some candidate where $P(w, c) \geq P(c, w)$, i.e., w has a stronger (or equally strong) path to c than the other way round. Assume that c can beat w by deleting the minimal number of candidates needed (with regards to the number necessary for making other candidates beat w). We claim that either those deleted candidates are direct neighbors of c, or there exists some other candidate

$c^* \in C$ for which we can reach the same goal by deleting equally many or even fewer candidates in the neighborhood of c^*. First, we define some notation. Let Del_w^c be the minimal number of removed candidates needed to make c beat w. Note that these candidates form a path-preserving vertex cut[5]. We say that x is *before* z if x is closer to w than z. Let $N^+(c)$ be the in-neighborhood of a candidate c, i.e., $N^+(c)$ contains all candidates with a direct edge to c. Finally, we define Ind_w^c to be the candidates, which belong to the connected component of c as induced by the vertex cut Del_w^c. Intuitively, Ind_w^c contains all candidates on stronger paths from w to c that are broken by deleting candidates and where the cut is before the candidate. Clearly, any $z^* \in Ind_w^c$ also beats w since first $P(z^*, c) \geq P(c, w)$ and, therefore, $P(z^*, w) \geq P(c, w)$, and secondly, no path from w to z^* with strength greater than $P(c, w)$ can exist. It follows that $|Del_w^{z^*}| \leq |Del_w^c|$. We distinguish the following two cases.

Case 1: $Ind_w^c = \emptyset$**.** Since Del_w^c is a minimal cut, we have $Del_w^c \subseteq N^+(c)$.

Case 2: $Ind_w^c \neq \emptyset$**.** Let $F = \{f \in Ind_w^c \mid N^+(f) \cap Del_w^c \neq \emptyset\}$ be the set of all candidates in the connected component of c, where we deleted some candidates in the in-neighborhood of c. On the one hand, if $|F| = 1$ we have a candidate $f \in F$ who also beats w by deleting Del_w^c. Since Del_w^c is minimal, it follows that $Del_w^f = Del_w^c$, and therefore, for a successful control action against w, it suffices to delete from $N^+(f)$. On the other hand, if $|F| > 1$, then $N^+(f) \cap Del_w^c = N^+(g) \cap Del_w^c$ for all $f, g \in F$. For a contradiction, assume there are two candidates $f, g \in F$ who do not share the same in-neighbors in Del_w^c. Deleting either $N^+(f) \cap Del_w^c$ or $N^+(g) \cap Del_w^c$ is sufficient to make the respective candidate beat w. Since $N^+(f) \cap Del_w^c \neq N^+(g) \cap Del_w^c$, we have a contradiction to Del_w^c being minimal. □

This result can be used to design an algorithm for Schulze-DCDC, which runs in polynomial time. Note that we will at times refer to candidates who could possibly beat d as *rivals of the despised candidate*.

Theorem 5. *Schulze-*DCDC *in the nonunique-winner model is solvable in polynomial time.*

Proof sketch. Consider an election (C, V) and a corresponding control instance $((C, V), d, \ell)$ of Schulze-DCDC. If the despised candidate d initially is a Schulze winner of (C, V), our goal is to find a candidate c with a stronger path to d than d has to c by deleting at most ℓ candidates.

First, if the despised candidate d is already not a Schulze winner of (C, V), return true. Otherwise, iterate over the set of candidates $C \setminus \{d\}$. For each candidate $c \in C \setminus \{d\}$, we check whether c is a possible rival of d. If d beats c directly, i.e., the edge from d to c is stronger than any path from c to d, we exclude c as a rival and move on to the next possible candidate. Additionally, if $P[c, d] = 0$, we also exclude c and move on. Otherwise, c is a possible rival of d and we move on to the deletion stage and initialize a deletion counter $ctr = 0$ for this candidate c. Next, for the graph G of all stronger paths from d to c, i.e., all paths where the strength of the path is greater than $P[c, d]$, we check whether deleting the in-neighbors $N_G^+(c)$ of c is possible within our deletion limit and accomplishes our goal of dethrowning d: We increment the deletion counter by $ctr{+}= |N_G^+(c)|$ and check whether $ctr > \ell$; if so, we move on to the next possible candidate; otherwise, we delete $N_G^+(c)$ from the original election. If d is not a Schulze winner of the election after deletion, return true. If d still wins, we have cut a path from c to d by deleting $N_G^+(c)$. Repeat the above steps until either the deletion limit is reached or d is not a winner of the election anymore. Finally, if there is no candidate such that deleting at most ℓ candidates makes this candidate win against d, return false.

The algorithm runs in polynomial time and correctness follows from Theorem 4. Note that it is not possible to easily extend this result to the unique-winner model. □

[5]See Menton and Singh (Menton and Singh 2013) for a definition of path-preserving vertex cut.

Future Research

We studied electoral control for Schulze and ranked pairs. After fixing a flaw in the proof of (Menton and Singh 2013, Thm. 2.2) for Schulze-CCDC, we turned to control by replacing candidates or voters and multimode control for Schulze and ranked pairs and solved a number of open problems for them. However, multiple variants of destructive control of the candidate set such as *candidate groups* (Erdélyi, Hemaspaandra, and Hemaspaandra 2015) as well as destructive control by adding (unique- and nonunique-winner model) or deleting candidates (unique-winner model) remain open for Schulze elections. Somewhat surprisingly, to the best of our knowledge, some cases of control by partition of candidates or voters are yet to be solved for ranked pairs elections.

Acknowledgements This work was supported in part by DFG grant RO-1202/21-1.

References

Bartholdi III, J. J.; and Orlin, J. B. 1991. Single Transferable Vote Resists Strategic Voting. *Social Choice and Welfare*, 8(4): 341–354.

Bartholdi III, J. J.; Tovey, C.; and Trick, M. 1989. The Computational Difficulty of Manipulating an Election. *Social Choice and Welfare*, 6(3): 227–241.

Bartholdi III, J. J.; Tovey, C.; and Trick, M. 1992. How Hard Is It to Control an Election? *Mathematical and Computer Modelling*, 16(8/9): 27–40.

Baumeister, D.; and Rothe, J. 2015. Preference Aggregation by Voting. In Rothe, J., ed., *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, Springer Texts in Business and Economics, chapter 4, 197–325. Heidelberg and Berlin, Germany: Springer-Verlag.

Berker, R. E.; Casacuberta, S.; Ong, C.; and Robinson, I. 2022. Obvious Independence of Clones. Technical Report arXiv:2210.04880v1 [cs.GT], ACM Computing Research Repository (CoRR).

Brill, M.; and Fischer, F. 2012. The Price of Neutrality for the Ranked Pairs Method. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, 1299–1305. Palo Alto, CA, USA: AAAI Press.

Conitzer, V.; Rognlie, M.; and Xia, L. 2009. Preference Functions That Score Rankings and Maximum Likelihood Estimation. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 109–115. Pasadena, CA, USA: IJCAI.

Conitzer, V.; and Sandholm, T. 2006. Nonexistence of Voting Rules That Are Usually Hard to Manipulate. In *Proceedings of the 21st National Conference on Artificial Intelligence*, 627–634. Palo Alto, CA, USA: AAAI Press.

Conitzer, V.; and Walsh, T. 2016. Barriers to Manipulation in Voting. In Brandt, F.; Conitzer, V.; Endriss, U.; Lang, J.; and Procaccia, A. D., eds., *Handbook of Computational Social Choice*, chapter 6, 127–145. Cambridge, UK: Cambridge University Press.

Erdélyi, G.; Hemaspaandra, E.; and Hemaspaandra, L. A. 2015. More Natural Models of Electoral Control by Partition. In *Proceedings of the 4th International Conference on Algorithmic Decision Theory*, 396–413. Heidelberg and Berlin, Germany: Springer-Verlag *Lecture Notes in Artificial Intelligence #9346*.

Erdélyi, G.; Neveling, M.; Reger, C.; Rothe, J.; Yang, Y.; and Zorn, R. 2021. Towards Completing the Puzzle: Complexity of Control by Replacing, Adding, and Deleting Candidates or Voters. *Journal of Autonomous Agents and Multi-Agent Systems*, 35(2): 41:1–41:48.

Faliszewski, P.; Hemaspaandra, E.; and Hemaspaandra, L. A. 2009. How Hard Is Bribery in Elections? *Journal of Artificial Intelligence Research*, 35: 485–532.

Faliszewski, P.; Hemaspaandra, E.; and Hemaspaandra, L. A. 2011. Multimode Control Attacks on Elections. *Journal of Artificial Intelligence Research*, 40: 305–351.

Faliszewski, P.; Hemaspaandra, E.; Hemaspaandra, L. A.; and Rothe, J. 2009. Llull and Copeland Voting Computationally Resist Bribery and Constructive Control. *Journal of Artificial Intelligence Research*, 35: 275–341.

Faliszewski, P.; and Rothe, J. 2016. Control and Bribery in Voting. In Brandt, F.; Conitzer, V.; Endriss, U.; Lang, J.; and Procaccia, A. D., eds., *Handbook of Computational Social Choice*, chapter 7, 146–168. Cambridge, UK: Cambridge University Press.

Gaspers, S.; Kalinowski, T.; Narodytska, N.; and Walsh, T. 2013. Coalitional Manipulation for Schulze's Rule. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems*, 431–438. www.ifaamas.org: IFAAMAS.

Gonzalez, T. F. 1985. Clustering to Minimize the Maximum Intercluster Distance. *Theoretical Computer Science*, 38: 293–306.

Hemaspaandra, E.; Hemaspaandra, L. A.; and Rothe, J. 2007. Anyone But Him: The Complexity of Precluding an Alternative. *Artificial Intelligence*, 171(5–6): 255–285.

Hemaspaandra, L. A.; Lavaee, R.; and Menton, C. 2013. Schulze and Ranked-Pairs Voting are Fixed-Parameter Tractable to Bribe, Manipulate, and Control. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems*, 1345–1346. www.ifaamas.org: IFAAMAS.

Lang, J.; Maudet, N.; and Polukarov, M. 2013. New Results on Equilibria in Strategic Candidacy. In *Proceedings of the 6th International Symposium on Algorithmic Game Theory*, 13–25. Heidelberg and Berlin, Germany: Springer-Verlag *Lecture Notes in Computer Science #8146*.

Loreggia, A.; Narodytska, N.; Rossi, F.; Venable, K. B.; and Walsh, T. 2015. Controlling Elections by Replacing Candidates or Votes (Extended Abstract). In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems*, 1737–1738. www.ifaamas.org: IFAAMAS.

McGarvey, D. C. 1953. A Theorem on the Construction of Voting Paradoxes. *Econometrica*, 21(4): 608–610.

Menton, C.; and Singh, P. 2012. Manipulation and Control Complexity of Schulze Voting. Technical Report arXiv: 1206.2111v1 [cs.GT], ACM Computing Research Repository (CoRR).

Menton, C.; and Singh, P. 2013. Control Complexity of Schulze Voting. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, 286–292. Beijing, China: AAAI Press/IJCAI.

Parkes, D.; and Xia, L. 2012. A Complexity-of-Strategic-Behavior Comparison between Schulze's Rule and Ranked Pairs. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, 1429–1435. Palo Alto, CA, USA: AAAI Press.

Schulze, M. 2011. A New Monotonic, Clone-Independent, Reversal Symmetric, and Condorcet-Consistent Single-Winner Election Method. *Social Choice and Welfare*, 36(2): 267–303.

Schulze, M. 2023. The Schulze Method of Voting. Technical Report arXiv:1804.02973v13 [cs.GT], ACM Computing Research Repository (CoRR).

Tideman, T. N. 1987. Independence of Clones as a Criterion for Voting Rules. *Social Choice and Welfare*, 4(3): 185–206.

Xia, L.; Zuckerman, M.; Procaccia, A. D.; Conitzer, V.; and Rosenschein, J. S. 2009. Complexity of Unweighted Coalitional Manipulation Under Some Common Voting Rules. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 348–353. Pasadena, CA, USA: IJCAI.