The Complexity of Cloning Candidates in Multiwinner Elections

Marc Neveling and Jörg Rothe
Institut für Informatik,
Heinrich-Heine-Universität Düsseldorf, Germany
{marc.neveling,rothe}@hhu.de

ABSTRACT

We initiate the study of cloning in multiwinner elections, focusing on single-transferable vote (STV), single-nontransferable vote (SNTV), bloc, k-Borda, t-approval-CC, and Borda-CC. Transferring the model of cloning due to Elkind et al. [15] from single-winner to multiwinner elections, we consider decision problems describing *possible* and *necessary* cloning in the zero-cost, the unit-cost, and the general-cost model and study their computational complexity. We show that, depending on the multiwinner voting rule and on the cost model chosen, some of these cloning problems are in P, some are NP-hard, and some of the latter (for which, in fact, already winner determination is NP-hard) are fixed-parameter tractable.

KEYWORDS

Computational social choice; Multiwinner elections; Cloning

1 INTRODUCTION

A common thread in computational social choice—see, e.g., the books edited by Brandt et al. [6] and Rothe [32]—is to study how the outcome of elections can be tampered with and how resistant voting rules are against such attempts in terms of computational complexity. The most thoroughly studied types of attack are manipulation (see, e.g., Conitzer and Walsh [12] and Baumeister and Rothe [4, Section 4.3.3]), electoral control (see, e.g., Faliszewski and Rothe [17] and Baumeister and Rothe [4, Section 4.3.4]), and bribery (see, e.g., Faliszewski and Rothe [17] and Baumeister and Rothe [4, Section 4.3.5]). On the other hand, relatively few papers have studied attacks by cloning candidates (see the related work below), and they are typically concerned with cloning in single-winner voting rules. We initiate the study of cloning in *multiwinner* elections, where the goal is not to elect a winner but to elect a *winning committee* of a certain size (see, e.g., the book chapter by Faliszewski et al. [18]). Multiwinner elections have various applications ranging from parliament elections over short-listing possible employees to selecting items to offer to a group of people (see Lu and Boutilier [26], Elkind et al. [14], and Skowron et al. [34] for more detailed descriptions of the mentioned settings). In each of those settings, cloning candidates might be beneficial for a candidate to be voted into the resulting committee. For instance, in a parliament election the campaign manager of a party, whose candidates may look like clones to ignorant voters, might be inclined to nominate a strategically chosen number of candidates for the party. Another application of cloning in multiwinner elections are movie recommender systems [21] in which a set of movies is recommended depending on the users' preferences: To influence the election result by spreading out and diminishing the support of a particular disliked movie, one might add to the election additional, very similar movies (e.g., other movies of the same genre or with a similar cast or by the same director).

In social choice theory, Tideman [35] introduced the notion of cloning and studied the *independence of clones* property for various voting rules. In particular, he showed that the single-winner variant of single-transferable vote (STV) is independent of clones. In a follow-up paper, Zavist and Tideman [36] studied independence of clones for the ranked pairs rule and presented a variant of ranked pairs that is even "completely independent of clones." Schulze voting is another widely used voting rule that is independent of clones [33]. In anonymous settings, such as the internet, voters may be tempted and able to cast their vote twice (or more often). This was the motivation for Conitzer [10] to introduce false-name manipulation as cloning of *voters* instead of candidates.[1] Recently, the independence of clones property was studied for the single-winner variant of STV with top-truncated votes by Ayadi et al. [1].

The paper by far most closely related to our work is due to Elkind et al. [15] (see also their follow-up paper [16]). They were the first to study how resistant single-winner voting rules are against cloning in terms of computational complexity. Adapting their model of cloning to multiwinner (rather than single-winner) elections, we consider decision problems describing *possible* cloning (where we ask whether a given candidate can become a member of a winning committee in at least one cloned multiwinner election, i.e., for at least one ordering of the clones) and *necessary* cloning (where we ask the same question for all cloned multiwinner elections, i.e., for all orderings of the clones), where the cloning costs are specified according to three cost models: zero cost, unit cost, and general cost. We study these problems in terms of their computational complexity and show that, depending on the multiwinner voting rule and on the cost model chosen, some of these cloning problems are in P, some are NP-hard, and some of the latter (for which, in fact, already winner determination is NP-hard) are in FPT, i.e., they are fixed-parameter tractable.

Organization. In Section 2, we present some background from social choice theory and multiwinner voting rules. In Section 3, we describe our model and define the problems to be studied in terms of their complexity. Section 4 contains our results and Section 5 our conclusions and some open problems.

---

[1]False-name manipulation [2, 30] has also been studied in cooperative game theory and the related property of duplication monotonicity [3, 24] in fair division.

2 PRELIMINARIES

A multiwinner election $E = (C, V, k)$ is defined by a set of candidates $C = \{c_1, \ldots, c_m\}$, a list of votes $V = (v_1, \ldots, v_n)$, and a committee size k. Votes are strict linear orders over the candidates and we write them each as a sequence of candidates, with the voter's preference strictly decreasing from left to right, so the leftmost (rightmost) candidate in a vote is most (least) preferred by this voter (e.g., if $C = \{a, b, c, d\}$, a vote $b\ a\ c\ d$ means that b is preferred to a, a to c, and c to d).

Given a multiwinner election (C, V, k), a multiwinner voting rule returns a nonempty family of size-k subsets of C, referred to as the *winning committees*. Given (C, V, k) and a fixed $t \geq 1$, the *t-approval score of a candidate* $a \in C$ is the number of votes in which a is ranked in the first t positions, and a's *Borda score* is the total number of points a scores in all votes of V, where a is rewarded with $m - i$ points whenever a is ranked in the ith position of a vote.

We consider the following multiwinner voting rules (with n voters and committee size k):

Single transferable vote (STV): Let $q = \lfloor n/(k+1) \rfloor + 1$ be the quota. Iteratively, if a candidate c is ranked first in at least q votes, add c to the winning committee and remove both c and q votes that rank c first from the multiwinner election, or else eliminate a candidate from the multiwinner election that is ranked first in the smallest number of votes. The iteration halts as soon as k candidates have been selected. To break ties between candidates we will use a predefined lexicographic tie-breaking order and ties between votes (i.e., when a candidate is ranked first in more than q votes but only q of those votes will be removed) are broken arbitrarily.[2]

Single nontransferable vote (SNTV): Choose k candidates with highest 1-approval score.

Bloc: Choose k candidates with highest k-approval score.

k-Borda: Choose k candidates with highest Borda score.

t-approval-CC (where CC stands for "Chamberlin–Courant" [9]): A voter v approves a committee if v ranks a committee member in the first t positions and disapproves it otherwise. The committee(s) with the most approvals from the voters win(s).

Borda-CC: Works similarly as t-approval-CC except that the voters assign to each committee the Borda score of its highest ranked member in their preferences.

Note that t-approval-CC and Borda-CC have an NP-hard winner determination problem [26, 29], though they are in FPT if parameterized by the number of candidates or voters [5].

For (C, V, k) a multiwinner election and candidates $c, d \in C$, let $score_{(C,V,k)}(c)$ denote the number of points c scores (according to t-approval or Borda which is always clear from the context) and let

$$dist_{(C,V,k)}(c, d) = score_{(C,V,k)}(c) - score_{(C,V,k)}(d)$$

denote the difference between the scores of c and d in (C, V, k). We sometimes omit the subscript (C, V, k) if it is clear from the context. If $S \subseteq C$ is a subset of the candidates, $\overrightarrow{S}$ in a vote denotes a ranking of these candidates in an arbitrary but fixed order and $\overleftarrow{S}$ denotes this ranking in reverse order. For example, for $C = \{a, b, c, d\}$ and $S = \{a, d\}$ and assuming the lexicographic order of candidates, $c \overrightarrow{S} b$ denotes the vote $c\ a\ d\ b$ and the vote $c \overleftarrow{S} b$ denotes $c\ d\ a\ b$.

---

[2]We cannot use "parallel-universe tie-breaking" [11] for STV since winner determination would then already be NP-hard.

3 MODEL AND PROBLEM DEFINITIONS

In this section, we will formalize how cloning is modeled for multiwinner elections. Let $E = (C, V, k)$ be a multiwinner election with $C = \{c_1, \ldots, c_m\}$ and $V = (v_1, \ldots, v_n)$. Let $K = (K_1, \ldots, K_m)$ with $K_i \geq 0$ be a vector, called a *cloning vector*. Intuitively, K_i means that the candidate c_i is cloned K_i times and c_i is replaced by her clones in the multiwinner election. If $K_i = 0$, the candidate c_i is not cloned and simply remains in the multiwinner election. Note that Elkind et al. [15] require that every candidate is cloned at least once, which is equivalent to our definition, but we feel it may be more natural and convenient if one can choose not to clone a candidate.

A multiwinner election $E_K = (C', V', k)$ is created by cloning from E via the cloning vector K if $C' = (C \setminus \{c_i \in C \mid K_i \geq 1\}) \cup \{c_i^{(j)} \mid 1 \leq j \leq K_i\}$ and $V' = (v_1', \ldots, v_n')$ with each $v_i' \in V'$ being a total order over C' that results from v_i by replacing cloned candidates in the vote v_i with their clones (i.e., for each clone c_i' of c_i, it holds that c_i' is preferred to $c_j \in C'$ in v_i' if and only if c_i is preferred to c_j (or her original candidate if c_j is a clone) in v_i).

Note that there can be several possible cloned multiwinner elections depending on how the clones of the same candidate are ordered in the votes. The goal of cloning a multiwinner election is to make a distinguished candidate (always called p) or one of p's clones a member of at least one winning committee. Regarding the ordering of clones of the same candidate in the votes, we use an optimistic and a pessimistic approach. In the optimistic setting, cloning via a cloning vector K is considered to be successful if the distinguished candidate (or one of her clones) is a member of a winning committee for *at least one* cloned multiwinner election via K. In the pessimistic setting, cloning via a cloning vector K is considered to be successful if the distinguished candidate (or one of her clones) is a member of a winning committee in *all* cloned multiwinner elections via K. Additionally, as is common in the literature, we adopt the so-called *nonunique-winner model* in which we assume a cloning action to be successful if the distinguished candidate is part of *at least one* winning committee instead of *all* winning committees, which would be required in the *unique-winner model*. Furthermore, we consider the cost of cloning candidates. In the *general-cost (GK) model*, for every candidate $c_i \in C$ there is a cost function $\rho_i : \mathbb{N} \to \mathbb{N}$ with $\rho_i(0) = \rho_i(1) = 0$ and for each $j, j' \in \mathbb{N}$ with $j < j'$ it holds that $\rho_i(j) \leq \rho_i(j')$. $\rho_i(j)$ is the cost of cloning the ith candidate j times and replacing her in all votes with her clones. There also is an integer B, called the budget. Additionally, we study two natural special cases of the general-cost model: The *unit-cost (UC) model* in which $\rho_i(j) = j - 1$ for all i and $j \geq 1$ (i.e., every additional clone costs one and there is a maximum number of additional clones), and a special case of the unit-cost model, the *zero-cost (ZK) model*, in which either the budget is set to infinity, or $\rho_i(j) = 0$ for all i and $j \geq 1$. In the latter cost model, since the budget is not a concern in this case, we seek to find out whether a successful cloning is even possible in the first place.

We can now define the decision problems we will consider. Let $\mathcal{R}$ be a multiwinner voting rule. In the problem $\mathcal{R}$-POSSIBLE-CLONING-GC, we are given a multiwinner election $E = (C, V, k)$, a cost function $\rho_i : \mathbb{N} \to \mathbb{N}$ for every $c_i \in C$, a distinguished candidate $p \in C$, and a budget B, and we ask whether there is a cloning vector $K = (K_1, \ldots, K_m)$ with $\sum_{c_i \in C} \rho_i(K_i) \leq B$ such that p (or one of its

Table 1: Overview of complexity results for various multiwinner voting rules

voting rule	parameter	Possible-Cloning			Necessary-Cloning		
		ZC	UC	GC	ZC	UC	GC
STV		NP-hard			coNP-hard		
SNTV		P			–		
Bloc		NP-hard			NP-hard		
k-Borda		P	P	NP-hard	NP-hard		
t-approval-CC	#candidates	?			FPT		
	#voters	FPT			FPT		
Borda-CC	#voters	?	?	W[1]-hard	?	?	W[1]-hard

clones) is in a winning committee under $\mathcal{R}$ in at least one cloned multiwinner election E_K resulting from E via K.

The problem $\mathcal{R}$-Necessary-Cloning-GC is defined analogously, except that we ask whether p ends up in a winning committee under $\mathcal{R}$ for all multiwinner elections E_K obtained from E by cloning via K.

If we use the unit-cost or the zero-cost model in this definition, we replace GC in the problem name by UC or ZC and omit the cost functions in the problem instances, and in the case of the zero-cost model we also omit the budget.

We assume the reader to be familiar with the basic notions of computational complexity theory, both in the classical branch (see, e.g., the books by Papadimitriou [28] and Rothe [31]) and the parameterized branch (see, e.g., the books by Downey and Fellows [13] and Niedermeier [27]). Since the zero-cost model is a special case of the unit-cost model, which in turn is a special case of the general-cost model, it holds that: $\mathcal{R}$-$\star$-Cloning-ZC reduces to $\mathcal{R}$-$\star$-Cloning-UC, which in turn reduces to $\mathcal{R}$-$\star$-Cloning-GC, where $\star \in \{\text{Possible}, \text{Necessary}\}$.

4 RESULTS

In this section, we present our results on the complexity of cloning in various multiwinner voting rules; see Table 1 for an overview. Question marks in this table indicate open problems and "–" means that influencing the outcome of a multiwinner election via this type of cloning and under this multiwinner voting rule is impossible.

4.1 STV

We show that possible cloning with zero cost is NP-hard for STV, even if the committee size is fixed to two.

Theorem 4.1. *STV-Possible-Cloning-ZC is NP-hard, even if $k = 2$.*

Proof. To prove this theorem, we will need the following observation and lemmas (proofs are omitted due to space constraints).

Observation 1. *Cloning a candidate does not change the plurality score of any other candidates or their clones.*

Lemma 4.2. *In an STV multiwinner election, the order in which candidates (or their last standing clones) are deleted from the multiwinner election in rounds where the quota is not reached cannot be changed by cloning those candidates.*

Lemma 4.3. *In an STV multiwinner election, a candidate in a winning committee that is always added last to its winning committees*

can be cloned without changing the outcome of the multiwinner election.

To show NP-hardness of STV-Possible-Cloning-ZC, we now reduce from the well-known NP-hard problem X3C (see, e.g., Garey and Johnson [20]) to STV-Possible-Cloning-ZC. Let $(X, \mathcal{S})$ with $X = \{x_1, \dots, x_{3s}\}$ and $\mathcal{S} = \{S_1, \dots, S_{3s}\}$ be a given X3C instance and assume that every $x_i \in X$ appears in exactly three elements of $\mathcal{S}$ (that this restriction of X3C is still NP-complete was shown by Gonzalez [22]). We also assume that $s \geq 3$ is even, which can be achieved by duplicating the instance. The set of candidates is $C = \{p, c, d, e, f\} \cup X \cup \mathcal{S} \cup B$, where $B = \{b_1, \dots, b_{3s}\}$ and p is the distinguished candidate. Set the committee size to $k = 2$. Since we are in the zero-cost model, the budget is set to infinity. For each $x_i \in X$, let $\mathcal{S}_{x_i} = \{S_j \in \mathcal{S} \mid x_i \in S_j\}$. We define V to consist of the votes shown in Table 2.

Table 2: List of votes V for the proof of Theorem 4.1

number	vote	for
$9\frac{s^2}{2} + 49\frac{s}{2} + 13$	$d\ e\ f\ p$	
1	$d\ S_i\ c\ p$	$1 \leq i \leq 3s$
$\frac{s}{2} + 1$	$S_i\ e\ p$	$1 \leq i \leq 3s$
$\frac{s}{2} + 2$	$S_i\ f\ p$	$1 \leq i \leq 3s$
$s + 5$	$x_i\ S_{x_i}\ c\ p$	$1 \leq i \leq 3s$
$\frac{s}{2} + 2$	$b_i\ S_i\ e\ p$	$1 \leq i \leq 3s$
$\frac{s}{2} + 2$	$b_i\ S_i\ f\ p$	$1 \leq i \leq 3s$
$4s + 8$	$p\ c$	
$4s + 7$	$c\ p\ e$	
$4s + 4$	$e\ p\ c$	
$4s + 4$	$f\ p\ c$	

We will break ties according to the linear order $\overrightarrow{B}\ \overrightarrow{C \setminus B}$.

To complete the proof of Theorem 4.1, we will now show that $(X, \mathcal{S})$ is a yes-instance of X3C if and only if p can be made a winner of at least one winning committee obtained from $(C, V, 2)$ by cloning, i.e., we have a yes-instance of STV-Possible-Cloning-ZC. Due to space constraints, however, we will only sketch the proof of the implication from left to right (and will then prove the converse direction in full detail): From left to right, suppose there is

an exact cover $\mathcal{S}'$. Clone d twice, and order them in such a way that the s votes of the form $d\, S_i\, p\, c$ for every $S_i \in \mathcal{S}'$ are not removed from the election when one clone of d is added to the winning committee. Then, p is later added to the winning committee.

For the converse direction, assume there is no exact cover. From Lemmas 4.2 and 4.3 we know that cloning candidates other than d has no effect on the outcome of the multiwinner election. Note that candidate d has s points more than needed to reach the quota and d will not gain any additional points before p is eliminated. If the clones of d are ordered in a way such that no clone reaches the quota and every clone has at least $4s + 9$ points then the multiwinner election proceeds as if d were not cloned and added to the winning committee up to the point in time when p is eliminated from the multiwinner election.

If there are clones with fewer than $4s + 9$ points, they will be eliminated before the elimination of p and transfer their points to other clones of d. If all clones with fewer than $4s + 9$ points are eliminated and there is still no clone who reaches the quota, we have the same situation as before where p will be eliminated. If at some point a clone of d reaches the quota (and p is still present in the multiwinner election), she will be added to the winning committee and all but up to s of her first-place votes will be removed, leaving s votes where d was in the first position in the original multiwinner election. Since q arbitrary first-place votes are removed if a clone of d has more first-place votes than the quota, we can definitely "save" some of those votes only by cloning d and assigning clones to the top of those votes that are not added to the winning committee. Note that if d is not cloned at all, d reaches the quota with s extra votes. Due to arbitrary tie-breaking of votes we might still be lucky and (at most) s votes of the form $d\, S_i\, c\, p$ are not removed from the election. Then we arrive at the same situation as below.

We will now show that it does not matter which s votes are prevented from being removed from the multiwinner election when a clone of d reaches the quota, since p will always be eliminated when there is no exact cover. Firstly, whenever a clone of d reaches the quota and is added to the winning committee, all remaining clones will be eliminated next, since they have at most s points and all other candidates have more than s points at any time. Secondly, saving votes of the form $d\, e\, f\, p\, \cdots$ from being removed is not advantageous for p, since she can beat e and f only much later in the multiwinner election (as can be seen in the original election). Also, the other votes that *can* be saved will give p additional points only if c is deleted earlier than p. Notice that in the original multiwinner election the candidates from $\mathcal{S}$ were eliminated immediately after d was added to the winning committee. By saving some votes of the form $d\, S_i\, c\, p\, \cdots$ we can save up to s candidates in $\mathcal{S}$ from being eliminated in the first $5s + 1$ rounds; let $\mathcal{S}'$ be the set of those candidates. Instead of the candidates from $\mathcal{S}$ without those up to s candidates, B and X can be eliminated earlier. Note that when candidates from B are eliminated, they are tieing the candidates from $\mathcal{S}'$ in points but we will see soon that we want the candidates from $\mathcal{S}'$ to be eliminated as late as possible for p to have a chance to survive longer.

Without candidates from B, the remaining candidates from $\mathcal{S}$ now have more points than p. Since we cannot prevent the candidates from X from being eliminated before c, those candidates will transfer their points to either c or a candidate from $\mathcal{S}'$ that is still

standing. To be precise, a candidate x_i will transfer her $s + 5$ points to a still-standing candidate from $S_{x_i} \cap \mathcal{S}'$ or to c if all candidates corresponding to members of S_{x_i} have already been eliminated.

If c gains points during the elimination of the candidates from X, c will have more points than p. Therefore, p only survives the round after the elimination of all candidates from X if for every x_i there is an $S_j \in \mathcal{S}'$ with $S_j \in S_{x_i}$ that is still present in the multiwinner election. Since $|\mathcal{S}'| \leq s$ and every $S_j \in \mathcal{S}$ is in exactly three subsets S_{x_i}, this is only possible if $\mathcal{S}'$ is an exact cover, which contradicts the assumption that there is none. ❑

Note that, by Lemma 4.2 and Lemma 4.3, influencing the result of the multiwinner election by cloning is impossible if $k = 1$. This is, in fact, not very surprising, since single-winner STV is independent of clones [35].

The reduction above can be modified to show that constructive control by adding candidates—see [4, 17] for its definition and an overview of results for it—is NP-hard for STV.

Regarding STV-Necessary-Cloning-ZC, we can show that it is coNP-hard (the proof is omitted due to space constraints). Notice that in contrast to the Possible-Cloning variant we cannot fix k here.

THEOREM 4.4. *STV-Necessary-Cloning-ZC is* coNP-*hard.*

Proof. To show coNP-hardness of STV-Necessary-Cloning-ZC, we now reduce from the complement of X3C to STV-Necessary-Cloning-ZC. Let $(X, \mathcal{S})$ be a given X3C instance, where $X = \{x_1, \ldots, x_{3s}\}$ and $\mathcal{S} = \{S_1, \ldots, S_{3s}\}$. Again, assume that every $x_i \in X$ appears in exactly three elements of $\mathcal{S}$ (recall the result by Gonzalez [22]). We also assume that $s \geq 3$, which can be achieved by duplicating the instance. The set of candidates is $C = \{p, r_1, r_2\} \cup X \cup \mathcal{S} \cup B \cup D \cup E \cup F$, where $B = \{b_1, \ldots, b_{3s}\}$, $D = \{d_1, \ldots, d_s\}$, $E = \{e_1, \ldots, e_{3s}\}$, $F = \{f_1, \ldots, f_{3s}\}$, and p is the distinguished candidate. Set the committee size to $k = s + 1$. Since we are in the zero-cost model, the budget is set to infinity. For each $x_i \in X$, let $S_{x_i} = \{S_j \in \mathcal{S} \mid x_i \in S_j\}$. We define V to consist of the votes shown in Table 3.

Table 3: List of votes V for the proof of Theorem 4.4

number	vote	for
$25s + 2$	$d_j\, r_1\, r_2\, S_1\, p$	$1 \leq j \leq s$
1	$d_j\, r_1\, r_2\, S_i\, p$	$1 \leq j \leq s$ and $1 \leq i \leq 3s$
2	$S_i\, e_i\, p$	$1 \leq i \leq 3s$
3	$b_i\, S_i\, f_i\, p$	$1 \leq i \leq 3s$
4	$x_i\, S_{x_i}\, r_1\, p$	$1 \leq i \leq 3s$
2	p	
1	$r_1\, p$	
1	$r_2\, r_1\, p$	
5	$e_i\, p$	$1 \leq i \leq 3s$
4	$f_i\, p$	$1 \leq i \leq 3s$

We will use the linear order $\overrightarrow{X}\, p\, r_1\, r_2\, \overrightarrow{B}\, \overrightarrow{S}\, \overrightarrow{D}\, \overrightarrow{F}\, \overrightarrow{E}$ to break ties. It does not matter how ties are broken if more than one candidate

reaches the quota, or which votes are removed from the multiwinner election if a candidate scores more points than the quota.

Let us analyze the multiwinner election $(C, V, s + 1)$ we have just constructed. Since $|V| = 54s + s(28s + 2) + 4$, the quota is

$$\left\lfloor \frac{54s + s(28s + 2) + 4}{s + 2} \right\rfloor + 1 = 28s + \left\lfloor \frac{4}{s + 2} \right\rfloor + 1 = 28s + 1.$$

Each candidate $d_j \in D$ reaches the quota with $28s + 2$ points and is added to the winning committee, and all but one vote $d_j \cdots$ for each $d_j \in D$ is removed from the multiwinner election. Since d_j is removed from each remaining vote, r_1 gains s points. In the following round, no one reaches the quota and r_2 is removed from the multiwinner election. In the next round, p and every candidate from $\mathcal{S}$ are tied for the lowest score, so p is eliminated due to the tie-breaking rule and is not part of the winning committee.

To complete the proof of Theorem 4.4, we will now show that $(X, \mathcal{S})$ is a no-instance of X3C if and only if p can be made part of at least one winning committee obtained from $(C, V, s + 1)$ by cloning, i.e., we have a yes-instance of STV-Necessary-Cloning-ZC.

From left to right, suppose there is no exact cover. We now show that there is a cloning vector such that for every possible ordering of clones p is part of a winning committee. Consider the cloning vector in which every candidate from D is cloned twice and consider three cases on how clones of a $d_j \in D$ can be ordered: (1) one clone reaches the quota and the other has a score of one, (2) one clone reaches the quota and the other has a score of zero (i.e., the ordering of clones is always the same for the votes where d_j was in the top position), and (3) both clones do not reach the quota. In the first two cases, the candidate who reaches the quota will be added to the winning committee and after all but one of her top position votes were removed from the multiwinner election, there is now a vote $d_j^{(2)} \ r_1 \ r_2 \ S_i \ p$ in which the other clone $d_j^{(2)}$ is in the top position and scores one point. In the third case, both clones score at least two points and the multiwinner election continues without adding any one of them to the winning committee. Notice that, in all three cases, r_1 and r_2 do not gain points and, after all clones of candidates from D who reach the quota were added to the winning committee, the remaining clones have score at most one. So, r_1 and r_2 are eliminated from the multiwinner election in the next two rounds and after that all second clones of candidates from the cases (1) and (2) as well. At some point during the following rounds, for each $d_j \in D$ whose clones are ordered according to case (3), one clone might be eliminated which would lead to the other clone reaching the quota in the next round. From the then not removed vote of the form $d_j \cdots$ either some $S_i \in \mathcal{S}$ or p gains a point. The latter would help p reaching the quota (but it is not needed), so we assume the worst case that some $S_i \in \mathcal{S}$ gains a point and that the clones from case (3) are eliminated or added to the winning committee now. Therefore, as soon as r_1 and r_2 and all clones of candidates from D are not part of the multiwinner election anymore, there is a subset $\mathcal{S}' \subseteq \mathcal{S}$ with $|\mathcal{S}'| \le s$ of candidates from $\mathcal{S}$ who gained at least one and up to s points from the removed clones of candidates from D. Then we have the following scores:

Candidate	p	$b_i \in B$	$e_i \in E$	$f_i \in F$	$x_i \in X$
Score	4	3	5	4	4

$S_i \in \mathcal{S} \setminus \mathcal{S}'$	$S_i \in \mathcal{S}'$
2	≥ 3

Therefore, no one reaches the quota in the following round, so all candidates from $\mathcal{S} \setminus \mathcal{S}'$ (transferring their points to candidates from $E' = \{e_i \in E \mid S_i \in \mathcal{S} \setminus \mathcal{S}'\}$) and B (transferring their points to candidates from $F' = \{f_i \in F \mid S_i \in \mathcal{S} \setminus \mathcal{S}'\}$ and $\mathcal{S}'$) are eliminated. Then the scores for the remaining candidates are as follows:

Candidate	p	$e_i \in E'$	$e_i \in E \setminus E'$	$f_i \in F'$
Score	4	7	5	7

$f_i \in F \setminus F'$	$x_i \in X$	$S_i \in \mathcal{S}'$
4	4	≥ 6

Due to the tie-breaking rule each candidate $x_j \in X$ is now eliminated transferring each of her four points to either a candidate from $\mathcal{S}_{x_j}$ if $\mathcal{S}_{x_j} \cap \mathcal{S}' \ne \emptyset$ or else to p. It follows that p does not gain points during this round only if $\mathcal{S}'$ is a cover of X, as then, for every candidate $x_j \in X$, there would be one candidate from $\mathcal{S}'$ sitting between x_j and p in those four votes of the form $x_j \ \mathcal{S}_{x_j} \ r_1 \ p$. Since $|\mathcal{S}'| \le s$, the cover $\mathcal{S}'$ must be an exact cover, which is not possible. Therefore, p gains at least four points from the elimination of candidates from X. Since p now has at least eight points and the score of candidates from F and E did not change, all those candidates are eliminated transferring their points to p. Note that $|E'| = |F'| = |\mathcal{S} \setminus \mathcal{S}'| \ge 2s$. Then the score of p is at least $8 + 3s(5 + 4) + (3 + 2)|\mathcal{S} \setminus \mathcal{S}'|27s + 5|\mathcal{S} \setminus \mathcal{S}'| + 8 \ge 37s + 8$. Therefore, p is added to the winning committee. Due to space constraints, we omit the proof of the direction from right to left. □

4.2 SNTV

By modifying a proof of Elkind et al. [15], we obtain:

Theorem 4.5. *SNTV-Possible-Cloning-GC is in* P.

Necessary cloning for SNTV (in any cost model) is impossible if p is not already in a winning committee, since we can order the clones such that one of the clones is in front of all other clones of her original candidate in all votes. Therefore, all but one clone of a candidate have zero points and the one clone has the same number of points as its original candidate in the initial multiwinner election. Therefore, if a candidate was part of a winning committee, then one of her clones is in a winning committee as well.

4.3 Bloc Voting

For bloc voting, we have NP-hardness results both for possible and necessary cloning in the zero-cost model. We omit the proof for possible cloning due to space constraints and present that for necessary cloning in detail here.

Theorem 4.6. *Bloc-Possible-Cloning-ZC is* NP-*hard, even if* $k = 2$.

Theorem 4.7. *Bloc-Necessary-Cloning-ZC is* NP-*hard, even if* $k = 2$.

Proof. For a fixed $t \ge 2$, t-approval-Necessary-Cloning-ZC was shown to be NP-hard by Elkind et al. [15]. We will reduce 2-approval-Necessary-Cloning-ZC to Bloc-Necessary-Cloning-ZC. Let $((C, V), p)$ be an instance of 2-approval-Necessary-Cloning-ZC. Set the committee size to $k = 2$. Therefore, bloc voting uses

2-approval scores. We create an additional candidate $w \notin C$ and a set D of $|V| + 1$ additional dummy candidates. Next, we create a list V' of $|V| + 1$ votes which have w in the first position and a dummy candidate from D in the second position such that every dummy candidate only scores one point from those new votes. The other candidates can be ordered arbitrarily. Furthermore, the new candidates are ordered last in all votes of V. We show that $((C, V), p)$ is a yes-instance of 2-approval-Necessary-Cloning-ZC if and only if $((C \cup D \cup \{w\}, V \cup V', 2), p)$ is a yes-instance of Bloc-Necessary-Cloning-ZC.

From left to right, assume that $((C, V), p)$ is a yes-instance of 2-approval-Necessary-Cloning-ZC. Then we can clone candidates from C such that p has the highest score in (C, V). Note that the score of p is larger than 1 and at most $|V|$. Thus we can clone candidates from C such that p has the second highest score in $(C \cup D \cup \{w\}, V \cup V', 2)$, since the candidates from C do not gain additional points from V', all additional dummy candidates score only one point and w scores $|V| + 1$ points which is a higher score than p has. Therefore, p is in a winning committee of $(C \cup D \cup \{w\}, V \cup V', 2)$.

For the converse direction, assume that $((C, V), p)$ is a no-instance of 2-approval-Necessary-Cloning-ZC. Then, whichever candidates of C we clone, p is never a winner of (C, V), which means that there is a candidate with a higher score than p. Therefore, p is always behind one candidate of C in $(C \cup D \cup \{w\}, V \cup V', 2)$ as well, since cloning w or dummy candidates does not change the allocation of points in V and no candidate of C gains additional points from the votes in V'. If p has the second-highest score of all candidates in C, it could still reach a winning committee if we could reduce the score of w by cloning her, but this is not possible since the voters of V' could order the clones of w such that one clone scores $|V'| = |V| + 1$ points, which is a higher score than any candidate in C can have. It follows that p cannot be in any winning committee of $(C \cup D \cup \{w\}, V \cup V', 2)$ if the order of clones cannot be controlled. ❏

4.4 k-Borda

Elkind et al. [15] proved that k-Borda-Possible-Cloning-GC is NP-hard for the single-winner version. This lower bound immediately transfers to the multiwinner variant of the problem. When restricted to unit costs, we can show that it is easy to solve (the proof is omitted due to space constraints).

Theorem 4.8. k-Borda-Possible-Cloning-UC is in P.

On the other hand, in the zero-cost model the problem becomes NP-hard for k-Borda, even for size-1 committees.

Theorem 4.9. k-Borda-Necessary-Cloning-ZC is NP-hard, even if $k = 1$.

Proof. We prove NP-hardness by reducing X3C to 1-Borda-Necessary-Cloning-ZC.

Given an X3C instance (X, S) with $X = \{x_1, \ldots, x_{3s}\}$ and $S = \{S_1, \ldots, S_{3s}\}$ (again, we assume that every $x_i \in X$ appears in exactly three elements of S), the candidate set is $C = \{p, a, d\} \cup X \cup S$ and V is defined to consist of the following votes:

(1) $7s + 1$ times a vote $a \, p \, \overrightarrow{X} \, S \, d$ and a vote $\overleftarrow{X} \, a \, p \, S \, d$.

(2) A vote $\overrightarrow{X} \, p \, S \, a \, d$ and a vote $\overleftarrow{X} \, p \, S \, a \, d$.

(3) For every $S_i \in S$ and for every $x_j \in S_i$, there is a vote $x_j \, S_i \, a \, p \, \overrightarrow{X \setminus \{x_j\}} \, S \setminus \{S_i\} \, d$ and a vote $x_j \, S_i \, a \, p \, \overleftarrow{X \setminus \{x_j\}} \, S \setminus \{S_i\} \, d$.

We also need some voters to control the point balances between p and every $x_i \in X$ and between p and a:

(4) 13 times a vote $a \, p \, \overrightarrow{X} \, S \, d$ and a vote $p \, a \, \overleftarrow{X} \, S \, d$.

(5) $9s$ times a vote $\overrightarrow{X} \, a \, p \, S \, d$ and a vote $\overleftarrow{X} \, p \, a \, S \, d$.

(6) For every $x_j \in X$, there are $2s + 4$ times a vote $\overleftarrow{X \setminus \{x_j\}} \, a \, p \, x_i \, d \, S$ and a vote $x_i \, d \, p \, a \, \overrightarrow{X \setminus \{x_j\}} \, S$.

(7) 8 times a vote $\overrightarrow{X} \, p \, a \, S \, d$ and a vote $p \, \overleftarrow{X} \, a \, S \, d$.

(8) 16 times a vote $\overrightarrow{X} \, p \, a \, S \, d$ and a vote $a \, d \, p \, \overleftarrow{X} \, S$.

We have the following point balances between p and the others:

$$
\begin{aligned}
dist_{(C,V,1)}(p, a) &= -(14s + 2) + (6s + 2) - 18s + 24s \\
&= -26s + 24s = -2s, \\
dist_{(C,V,1)}(p, x_i) &= -(7s + 1) - (3s + 1) - 18 + 3s(9s - 3) \\
&\quad -(9s - 13)(3s + 2) - (2s + 4) = 2, \\
dist_{(C,V,1)}(p, S_i) &> 6, \text{ and } dist_{(C,V,1)}(p, d) > 0.
\end{aligned}
$$

Lemma 4.10. In the constructed election, if a candidate from $C \setminus S$ is cloned more than once, p or all clones of p lose the election.

Lemma 4.11. In the constructed election, cloning a candidate $S_i \in S$ twice changes the point balances between p and the other candidates in the following way:

(1) p loses at most 6 points on both clones of S_i,
(2) p gains 2 points on a,
(3) p loses 2 points on each $x_j \in S_i$,
(4) p does not gain or lose points on any $x_j \in X \setminus S_i$,
(5) p gains points on d, and
(6) p never loses points on candidates in $S \setminus \{S_i\}$.

The proofs of Lemmas 4.10 and 4.11 are omitted due to space constraints. Equipped with these two lemmas, we now show that (X, S) is a yes-instance of X3C if and only if (C, V) is a yes-instance of 1-Borda-Necessary-Cloning-ZC.

From left to right, suppose there is an exact cover S'. Clone every $S_i \in S'$ twice (i.e., the original candidate S_i is substituted by a clone and there is an additional clone of S_i). From Lemma 4.11 and the point balances in the original election it follows that p is now tieing a and every $x_i \in X$ in points and beats every other candidate. Therefore, p is a winner of the election.

From right to left, suppose we can make p a winner of the election by cloning candidates. From Lemma 4.10 it follows that we must clone candidates from S to make p not lose the election immediately. Adding an additional clone of any $S_i \in S$ to the election improves p's point balance with a by 2 points and worsens p's point balance with all $x_j \in S_i$ by 2 points. Considering the point balances before cloning any candidates, it follows that we may only clone each $S_i \in S$ at most twice (which means adding an additional clone of $S_i \in S$ to the election), as otherwise p would be beaten by all $x_j \in S_i$. Furthermore, we need to add at least k additional clones of candidates from S for p to at least tie a. Therefore, there exists an exact cover of X in S. ❏

Since 1-Borda is equivalent to the single-winner variant of k-Borda we also showed that Necessary-Cloning-ZC is NP-hard for single-winner Borda. The complexity of this problem was left open by Elkind et al. [15].

4.5 t-approval-CC

As winner determination for CC multiwinner voting rules is NP-hard, all considered problems are trivially NP-hard for those rules as well. We will now show, however, that t-approval-CC-Necessary-Cloning-GC is fixed-parameter tractable when parameterized by the number of either candidates or voters. The following lemma (the proof of which is omitted due to space constraints) will be helpful in the proofs of Theorems 4.13 and 4.14, the former of which is presented here while the latter is again omitted due to space constraints.

Lemma 4.12. *Given a multiwinner election (C, V, k) and a candidate $p \in C$, if we can make p be a member of a winning committee under t-approval-CC and for all possible orderings of clones, we can do so by cloning candidates up to t times.*

Theorem 4.13. *For a fixed $t \geq 2$, t-approval-CC-Necessary-Cloning-GC is in FPT when parameterized by the number of candidates.*

Proof.   Adapting the FPT-algorithm by Bredereck et al. [8] for t-approval-CC-Shift Bribery and using Lemma 4.12 we obtain an FPT-algorithm that solves the problem. Given an instance of t-approval-CC-Necessary-Cloning-GC with m candidates and n voters, iterate over all possible cloning vectors $(K_1, \ldots, K_m)$ with $K_i \leq t$ for all $1 \leq i \leq m$ that are feasible within the budget B. For each such cloning vector, iterate over all committees W in a cloned multiwinner election via K that preclude p or any clone of p. For each combination of cloning vector K and committee W, solve the following integer linear program (ILP). Let $m' \leq mt$ be the number of candidates in a cloned multiwinner election via K. There are $m!$ different types of votes in the original multiwinner election and $m'!$ different types of votes in any cloned multiwinner election via K. We order them arbitrarily and associate with each $i \in [m!]$ and each $j \in [m'!]$ the ith and jth vote type of the original and cloned multiwinner election, where $[a]$ is the set of integers less than or equal to an integer a. We then create an integer variable $S_{i,j}$ for each pair of vote types. $S_{i,j}$ represents the number of votes that had type i in the original multiwinner election and then have type j in the cloned multiwinner election after all partial votes were extended to complete votes. With n_i being the number of votes of type i in the original multiwinner election, we create the constraint $\sum_{j \in [m'!]} S_{i,j} = n_i$ for every $i \in [m!]$ to ensure that the number of votes stays the same in the cloned election. Next, we introduce a constraint $\sum_{i \in [m!], j \in [m'!]} S_{i,j} \cdot \text{feas}(i, j) = 0$ that ensures that it is possible to transform a vote of type i in the original multiwinner election to a vote of type j in the cloned multiwinner election. Here, we use a boolean variable $\text{feas}(i, j)$, which is zero if a vote of type $i \in [m!]$ can be transformed to a vote of type $j \in [m']$, and is one otherwise. We now create integer variables N_j for each $j \in [m'!]$ which describe the number of votes of type j in the cloned multiwinner election: $\sum_{i \in [m!]} S_{i,j} = N_j$. Then we have to make sure that the committee W beats all committees

that contain p or clones of p. For a committee C' and vote type i in the cloned multiwinner election, denote by $\omega(i, C')$ the score that a vote of type i assigns to the committee C'. Then, for each committee W' containing p or clones of p, we create the constraint: $\sum_{i \in [m'!]} \omega(i, W) \cdot N_i > \sum_{i \in [m'!]} \omega(i, W') \cdot N_i$.

The ILP tells us if there is any ordering of clones such that W beats every committee containing p or clones of p. If the ILP is not solvable for every committee W, there is a cloning vector such that in every cloned multiwinner election via this cloning vector there always is a committee containing p or a clone of p among the winning committees for all orderings of clones, so output accept. If all cloning vectors have been iterated over and there always is some ordering of clones such that a committee not containing p or clones of p beats all committees containing p or clones of p in a cloned multiwinner election, output reject. Due to Lemma 4.12 we only need to check cloning vectors in which every component is at most t. Additionally, $\text{feas}(i, j)$ and $\omega(i, C')$ can be precomputed in FPT before the ILP is solved.

Regarding the runtime, the ILP will be called at most $t^m \cdot 2^{mt}$ times and can be solved in FPT due to the famous result by Lenstra Jr. [25] (which was improved by Kannan [23] and by Fredman and Tarjan [19]) that ILPs can be solved in FPT with respect to the number of integer variables as the parameter.   ❑

Theorem 4.14. *For a fixed $t \geq 2$, t-approval-CC-Necessary-Cloning-GC is in FPT when parameterized by the number of voters.*

Next, we turn to t-approval-CC-Possible-Cloning-GC. We cannot use Lemma 4.12 for this problem, as it may be necessary to clone a candidate more than t times, since the order of clones may be chosen freely.

Example 4.15. Let $t = 1$ (i.e., we consider 1-approval-CC), $C = \{p, c_1, c_2\}$ and V consist of the following voters:

- one vote $p \, \cdots$,
- n_1 votes $c_1 \, \cdots$ for some $n_1 > 1$, and
- n_2 votes $c_2 \, \cdots$ for some $n_2 > 1$.

If $k = 2$, we can make p be part of a winning committee only by cloning c_1 at least $n_1 > t$ times or c_2 at least $n_2 > t$ times and by assigning a different clone of c_1 (respectively of c_2) to the top position of each of her first-ranked votes.

However, with the notion of *relevant candidates* we can show that the problem is in FPT when it is parameterized by the number of voters. The proof of Theorem 4.16 is omitted here due to space constraints.

Theorem 4.16. *For a fixed $t \geq 2$, t-approval-CC-Possible-Cloning-GC is in FPT when parameterized by the number of voters.*

4.6   Borda-CC

We will show that Borda-CC-Possible-Cloning-GC is $W[1]$-hard even for committees of size $k = 1$ (in which case Borda-CC is just single-winner Borda) when parameterized by the number of voters.

Theorem 4.17. *Borda-CC-Possible-Cloning-GC is $W[1]$-hard when parameterized by the number of voters, even if the committee size is one and there are only two different values of costs.*

Proof. We prove $W[1]$-hardness by providing a parameterized reduction from the problem Multicolored-Independent-Set: Given an undirected graph $G = (V(G), E(G))$, an integer f, and a partition of $V(G)$ into f sets $W_1, \ldots, W_f$, does there exist an independent set $X \subseteq V(G)$ (i.e., the induced subgraph of G restricted to X has no edges) that contains exactly one vertex of every set W_i, $1 \le i \le f$? Multicolored-Independent-Set is $W[1]$-hard when parameterized by the number of colors [13].

Let $(G, f, (W_1, \ldots, W_f))$ be a Multicolored-Independent-Set instance. We may assume that the number of vertices for each color is the same (so $|V(G)| = \ell \cdot f$ for some $\ell \ge 1$) and that there are no edges between vertices with the same color. For $v \in V(G)$, denote by $E(v)$ the set of edges incident to v and by $d(v)$ the degree of v. For each color i, $1 \le i \le f$, denote by $\delta(i)$ the sum of degrees of vertices with color i, and let $\Delta = \sum_{1 \le i \le f} \delta(i)$.

From $(G, f, (W_1, \ldots, W_f))$ we will now construct a Borda-CC-Possible-Cloning-GC instance. Let $C = \{p\} \cup V(G) \cup E(G) \cup H \cup D_1 \cup D_2$ with $H = \{h_1, \ldots, h_f\}$ and D_1 and D_2 being sets of dummy candidates whose sizes we will define later. For a color i, $1 \le i \le f$, let $W_i = \{v_1^{(i)}, \ldots, v_\ell^{(i)}\}$ and for a subset $X \subseteq V(G)$, let $G \setminus X$ be the graph G without vertices (and incident edges) of X. Define V to consist of these votes:

(1) For every color i, with $1 \le i \le f$, there are two votes:

$$p\ h_i\ \overrightarrow{E(v_1^{(i)})}\ v_1^{(i)}\ \cdots\ \overrightarrow{E(v_\ell^{(i)})}\ v_\ell^{(i)}\ \overrightarrow{E(G \setminus W_i)}\ \overrightarrow{V(G) \setminus W_i}\ \overrightarrow{H \setminus \{h_i\}}\ D_2\ D_1,$$

$$p\ h_i\ \overleftarrow{E(v_\ell^{(i)})}\ v_\ell^{(i)}\ \cdots\ \overleftarrow{E(v_1^{(i)})}\ v_1^{(i)}\ \overleftarrow{E(G \setminus W_i)}\ \overleftarrow{V(G) \setminus W_i}\ \overleftarrow{H \setminus \{h_i\}}\ D_2\ D_1.$$

(2) Two votes: $p\ \overrightarrow{H}\overrightarrow{D_2}\ \overrightarrow{E(G)}\overrightarrow{D_1}\ \overrightarrow{V(G)}$ and $\overleftarrow{E(G)}\overleftarrow{D_1}\ \overleftarrow{H}\overleftarrow{D_2}\ p\ \overleftarrow{V(G)}$.

To determine the number of dummy candidates needed, let us consider the point balances between p and candidates $h_i \in H$ and $e_j \in E(G)$ from the votes in the first group:

$$
\begin{aligned}
dist(p, h_i) &= 2 + (f-1)(2(E(G) + V(G)) + f + 2), \\
dist(p, e_j) &= 4 + 2(\ell - 1) + (f - 2)(2\ell + E(G) + 3) + \Delta.
\end{aligned}
$$

Then we set D_2 to contain $dist(p, h_i) + 2(f-1)$ candidates and D_1 to contain $dist(p, e_j) + 2(f-2) + 1$ candidates. Let $B = f$. Regarding the price functions, for every $v \in V(G)$ let the cost of cloning v twice be one and cloning her more than twice be $B + 1$. Then let the cost of cloning any other candidate more than once be $B + 1$. Finally, let $k = 1$. It is easy to see that we will only need to worry about the scores of p, of candidates from H, and of candidates $E(G)$, since p beats all other candidates even if candidates from $V(G)$ are cloned. For $h_i \in H$ and $e_j \in E(G)$, p is trailing behind h_i with $2(f-1)$ points and behind e_j with $2(f-2) + 1$ points. We will now show that $(G, f, (W_1, \ldots, W_f))$ is a yes-instance of Multicolored-Independent-Set if and only if the above constructed instance of Borda-CC-Possible-Cloning-GC is a yes-instance.

From left to right, suppose there is multicolored independent set $X \subseteq V(G)$. Clone every $v \in X$ twice (i.e., the original candidate v is substituted by a clone and there is an additional clone of v). Let i be the color of a $v \in X$ (i.e., $v \in W_i$). From the additional clone of v the candidate p gains two points on every candidate $H \setminus \{h_i\}$. Since $|V'| = h$ and each candidate in X has a different color p is now tied with every candidate in H. Since the vertex candidates cloned are an independent set for each $e = \{v, v'\}$, at least one of v and v' (say v) was not cloned. If v is of color i then there is another vertex

candidate of color i that was cloned (since X contains a vertex of every color), so p gained one point on e, and from the cloned vertex candidates that were not of the colors of v and v' candidate p gained $2(f - 2)$ points, so p at least ties e. Therefore, p now ties or beats all candidates from H and $E(G)$ and wins the multiwinner election.

From right to left, suppose there is no multicolored independent set. We can clone at most f vertex candidates twice. They must be of different colors each and we need to clone f vertex candidates or else p cannot beat all candidates from H. Let us analyze how a cloned vertex candidate $v \in V(G)$ with color i affects the points balance between p and the edge candidates in $E(G)$: (1) p gains zero points on edge candidates in $E(G)$, (2) p gains one point on edge candidates who were incident to vertices of $W_i \setminus \{v\}$ in G, and (3) p gains two points on all other edge candidates.

Since there is no multicolored independent set of size f, in each $X \subseteq V(G)$ with $|X| = f$ and each $v \in X$ having a different color, there must be $v, v' \in X$ such that $e = \{v, v'\} \in E(G)$. Assume the candidates in X were cloned twice. Since v and v' were cloned and no other candidate with the colors of v and v' were cloned, p could not gain any points on e from the cloning of v and v'. Although p gains $2(f - 2)$ points on e from the cloning of candidates $X \setminus \{v, v'\}$, e still beats p by one point. So, p cannot win the multiwinner election. ❑

Since in the reduction above the ordering of clones did not matter, the following holds as well.

Corollary 4.18. *Borda-CC-Necessary-Cloning-GC is* $W[1]$*-hard when parameterized by the number of voters, even if* $k = 1$.

5 CONCLUSIONS AND OPEN PROBLEMS

We have initiated the study of cloning in various well-known multiwinner elections. Our complexity results are summarized in Table 1. They imply that cloning is intractable in general and is tractable only for simple multiwinner voting rules (SNTV) or a few restricted cases (e.g., k-Borda-Possible-Cloning-ZC/UC). Studying the parameterized complexity of the related problems might be fruitful since cloning for more involved voting rules (such as t-approval-CC) can be fixed-parameter tractable, even though that is not necessarily so (e.g., not for Borda-CC).

There are a number of interesting open problems. Specifically, the parameterized complexity of possible cloning in t-approval-CC for #*candidates* (rather than #voters) remains open in all cost models, and so does that of possible and necessary cloning in Borda-CC in the zero-cost and unit-cost models for #voters and in all cost models for #candidates. Of course, there are many more multiwinner voting rules than those studied here (see the book chapter by Faliszewski et al. [18] for an overview), and we propose to extend to them the study initiated here.

Further possible research directions are to study additional cost models such as *all-or-nothing* cost-functions, as was done by Bredereck et al. [7] for Shift-Bribery, and to further explore the parameterized complexity for problems that are NP-hard.

REFERENCES

[1] M. Ayadi, N. Ben Amor, J. Lang, and D. Peters. 2019. Single Transferable Vote: Incomplete Knowledge and Communication Issues. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, 1288–1296.

[2] H. Aziz, Y. Bachrach, E. Elkind, and M. Paterson. 2011. False-Name Manipulations in Weighted Voting Games. *Journal of Artificial Intelligence Research* 40 (2011), 57–93.

[3] D. Baumeister, S. Bouveret, J. Lang, N. Nguyen, T. Nguyen, J. Rothe, and A. Saffidine. 2017. Positional Scoring-Based Allocation of Indivisible Goods. *Journal of Autonomous Agents and Multi-Agent Systems* 31, 3 (2017), 628–655.

[4] D. Baumeister and J. Rothe. 2015. Preference Aggregation by Voting. In *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, J. Rothe (Ed.). Springer-Verlag, 197–325.

[5] N. Betzler, A. Slinko, and J. Uhlmann. 2013. On the Computation of Fully Proportional Representation. *Journal of Artificial Intelligence Research* 47 (2013), 475–519.

[6] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia (Eds.). 2016. *Handbook of Computational Social Choice.* Cambridge University Press.

[7] R. Bredereck, J. Chen, P. Faliszewski, A. Nichterlein, and R. Niedermeier. 2016. Prices Matter for the Parameterized Complexity of Shift Bribery. *Information and Computation* 251 (2016), 140–164.

[8] R. Bredereck, P. Faliszewski, R. Niedermeier, and N. Talmon. 2016. Complexity of Shift Bribery in Committee Elections. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence.* AAAI Press, 2452–2458.

[9] J. Chamberlin and P. Courant. 1983. Representative Deliberations and Representative Decisions: Proportional Representation and the Borda Rule. *The American Political Science Review* 77, 3 (1983), 718–733.

[10] V. Conitzer. 2008. Anonymity-proof Voting Rules. In *Proceedings of the 4th International Workshop on Internet & Network Economics.* Springer-Verlag *Lecture Notes in Computer Science #5385*, 295–306.

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

[12] V. Conitzer and T. Walsh. 2016. Barriers to Manipulation in Voting. In *Handbook of Computational Social Choice*, F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia (Eds.). Cambridge University Press, 127–145.

[13] R. Downey and M. Fellows. 2013. *Parameterized Complexity* (2nd ed.). Springer-Verlag.

[14] E. Elkind, P. Faliszewski, P. Skowron, and A. Slinko. 2017. Properties of Multiwinner Voting Rules. *Social Choice and Welfare* 48, 3 (2017), 599–632.

[15] E. Elkind, P. Faliszewski, and A. Slinko. 2011. Cloning in Elections: Finding the Possible Winners. *Journal of Artificial Intelligence Research* 42 (2011), 529–573.

[16] E. Elkind, P. Faliszewski, and A. Slinko. 2012. Clone Structures in Voters' Preferences. In *Proceedings of the 13th ACM Conference on Electronic Commerce.* ACM Press, 496–513.

[17] P. Faliszewski and J. Rothe. 2016. Control and Bribery in Voting. In *Handbook of Computational Social Choice*, F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia (Eds.). Cambridge University Press, 146–168.

[18] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. 2017. Multiwinner Voting: A New Challenge for Social Choice Theory. In *Trends in Computational Social Choice*, U. Endriss (Ed.). AI Access Foundation, 27–47.

[19] M. Fredman and R. Tarjan. 1987. Fibonacci Heaps and Their Uses in Improved Network Optimization Algorithms. *J. ACM* 34, 3 (1987), 596–615.

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

[21] S. Ghosh, M. Mundhe, K. Hernandez, and S. Sen. 1999. Voting for Movies: The Anatomy of Recommender Systems. In *Proceedings of the 3rd Annual Conference on Autonomous Agents.* ACM Press, 434–435.

[22] T. Gonzalez. 1985. Clustering to Minimize the Maximum Intercluster Distance. *Theoretical Computer Science* 38 (1985), 293–306.

[23] R. Kannan. 1987. Minkowski's Convex Body Theorem and Integer Programming. *Mathematics of Operations Research* 12, 3 (1987), 415–440.

[24] B. Kuckuck and J. Rothe. 2019. Duplication Monotonicity in the Allocation of Indivisible Goods. *AI Communications* 32, 4 (2019), 253–270.

[25] H. Lenstra Jr. 1983. Integer Programming with a Fixed Number of Variables. *Mathematics of Operations Research* 8, 4 (1983), 538–548.

[26] T. Lu and C. Boutilier. 2011. Budgeted Social Choice: From Consensus to Personalized Decision Making. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence.* AAAI Press/IJCAI, 280–286.

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

[28] C. Papadimitriou. 1995. *Computational Complexity* (second ed.). Addison-Wesley.

[29] A. Procaccia, J. Rosenschein, and A. Zohar. 2008. On the Complexity of Achieving Proportional Representation. *Social Choice and Welfare* 30, 3 (2008), 353–362.

[30] A. Rey and J. Rothe. 2014. False-Name Manipulation in Weighted Voting Games is Hard for Probabilistic Polynomial Time. *Journal of Artificial Intelligence Research* 50 (2014), 573–601.

[31] J. Rothe. 2005. *Complexity Theory and Cryptology. An Introduction to Cryptocomplexity.* Springer-Verlag.

[32] J. Rothe (Ed.). 2015. *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division.* Springer-Verlag.

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

[34] P. Skowron, P. Faliszewski, and J. Lang. 2016. Finding a Collective Set of Items: From Proportional Multirepresentation to Group Recommendation. *Artificial Intelligence* 241 (2016), 191–216.

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

[36] T. Zavist and N. Tideman. 1989. Complete Independence of Clones in the Ranked Pairs Rule. *Social Choice and Welfare* 6, 2 (1989), 167–173.