Ties in Multiwinner Approval Voting

Łukasz Janeczko , **Piotr Faliszewski**
AGH University, Poland
{ljaneczk,faliszew}@agh.edu.pl

Abstract

We study the complexity of deciding if there is a tie in a given approval-based multiwinner election, as well as the complexity of counting tied winning committees. We consider a family of Thiele rules, their greedy variants, Phragmén's sequential rule, and Method of Equal Shares. For most cases, our problems are computationally hard, but for sequential rules we find an FPT algorithm for discovering ties (parameterized by the committee size). We also show experimentally that in elections of moderate size ties are quite frequent.

1 Introduction

In an approval-based multiwinner election, a group of voters expresses their preferences about a set of candidates—i.e., each voter indicates which of them he or she approves—and then, using some prespecified rule, the organizer selects a winning committee (a fixed-size subset of the candidates). Multiwinner elections can be used to resolve very serious matters—such as choosing a country's parliament—or rather frivolous ones—such as choosing the tourist attractions that a group of friends would visit—or those positioned anywhere in between these two extremes—such as choosing a department's representation for the university senate. In large elections, one typically does not expect ties to occur (although surprisingly many such cases are known[1]), but for small and moderately sized ones the issue is unclear (whenever we speak of a size of an election, we mean both the number of candidates and the number of voters). While perhaps a group of friends may manage to not spoil their holidays upon discovery that they were as willing to visit one monument as another, a person not selected for a university senate due to a tie may be quite upset, especially if this tie is discovered after announcing the results. To address such possibilities, we study the following three issues:

1. We consider the complexity of detecting if two or more committees tie under a given voting rule. While for most rules this problem turns out to be intractable, for many settings we find practical solutions (in most cases it is either possible to use a natural integer linear programming trick or an FPT algorithm that we provide).
2. We consider the complexity of counting the number of winning committees. We do so, because being able to count winning committees would be helpful in sampling them uniformly. Unfortunately, in this case we mostly find hardness and hardness of approximation results.
3. We generate a number of elections, both synthetic and based on real-life data, and evaluate the frequency of ties. It turns out to be surprisingly high.

We focus on a subfamily of Thiele rules [Thiele, 1895; Aziz *et al.*, 2015; Lackner and Skowron, 2021] that includes the multiwinner approval rule (AV), the approval-based Chamberlin–Courant rule (CCAV), and the proportional approval voting rule (PAV), as well as on their greedy variants. We also study satisfaction approval voting (SAV), the Phragmén rule, and Method of Equal Shares (MEqS). This set includes rules appropriate for selecting committees of individually excellent candidates (e.g., AV or SAV), diverse committees (e.g., CCAV or GreedyCCAV), or proportional ones (e.g., PAV, GreedyPAV, Phragmén, or MEqS); see the works of Elkind *et al.* [2017] and Faliszewski *et al.* [2017] for more details on classifying multiwinner rules with respect to their application. We summarize our results in Table 1.

The issue of ties and tie-breaking has already received quite some attention in the literature, although typically in the context of single-winner voting. For example, Obraztsova and Elkind [2011] and Obraztsova *et al.* [2011] consider how various tie-breaking mechanisms affect the complexity of manipulating elections, Freeman *et al.* [2015] study different tie-breaking schemes, such as parallel-universes tie-breaking and randomized ones, in single-winner voting (mainly for STV), and recently Xia [2021] has made a breakthrough in studying the probability of ties in large, randomly-generated single-winner elections. Xia [2022] also developed a novel tie-breaking mechanism, albeit for a somewhat different setting than ours. Finally, Conitzer *et al.* [2009] have shown that deciding if a candidate is a tied winner in an STV election is NP-hard. While STV is not an approval-based rule and they focused on the single-winner setting, many of our results are in a similar spirit. Omitted proofs are in the full version of the paper [Janeczko and Faliszewski, 2023].

[1] https://en.wikipedia.org/wiki/List_of_close_election_results

Rule	UNIQUE-COMMITTEE	#WINNING-COMM.
AV	P	P
SAV	P	P
CCAV	coNP-h., coW[1]-h.(k)	#P-h., #W[1]-h.(k)
PAV	coNP-h., coW[1]-h.(k)	#P-h., #W[1]-h.(k)
GreedyCCAV	coNP-com., FPT(k)	#P-h., #W[1]-h.(k)
GreedyPAV	coNP-com., FPT(k)	#P-h., #W[1]-h.(k)
Phragmén	coNP-com., FPT(k)	#P-h., #W[1]-h.(k)
MEqS (Ph. 1)	coNP-com., FPT(k)	#P-h.

Table 1: Our complexity results. The coNP-completeness results regarding {GreedyCCAV, GreedyPAV, Phragmén}-UNIQUE-COMMITTEE follow from the work of Faliszewski *et al.* [2022] The polynomial-time algorithms for AV and SAV are folk knowledge.

2 Preliminaries

By $\mathbb{R}_+$ we denote the set of nonnegative real numbers. For each integer t, we write $[t]$ to mean $\{1, \dots, t\}$. We use the Iverson bracket notation, i.e., for a logical expression F, we interpret $[F]$ as 1 if F is true and as 0 if it is false. Given a graph G, we write $V(G)$ to denote its set of vertices and $E(G)$ to denote its set of edges. For a vertex v, by $d(v)$ we mean its degree (i.e., the number of edges that touch it).

An election $E = (C, V, A)$ consists of a set of candidates $C = \{c_1, \dots, c_m\}$ and a collection of voters $V = (v_1, \dots, v_n)$, where each voter v_i has a set $A(v_i) \subseteq C$ of candidates that he or she approves. We refer to this set as v_i's approval set or v_i's vote, interchangeably. We typically omit writing A (as it is clear from the context) and use a shortened notation $E = (C, V)$. By a small abuse of notation, for a candidate c we write $A(c)$ to denote the set of voters that approve him or her. A multiwinner voting rule f is a function that given an election $E = (C, V)$ and committee size $k \in [|C|]$ outputs a family of size-k subsets of C, i.e., a family of winning committees (denoted as $f(E, k)$). Below we describe the rules that we focus on.

Let $E = (C, V)$ be an election and let k be the committee size. Under the multiwinner approval rule (AV), each voter assigns a single point to each candidate that he or she approves and winning committees consist of k candidates with the highest scores. Satisfaction approval voting (SAV) proceeds analogously, except that each voter $v \in V$ assigns $1/|A(v)|$ points to each candidate he or she approves. In other words, under AV each voter can give a single point to each approved candidate, but under SAV he or she needs to split a single point equally among them.

Next we consider the class of Thiele rules, defined originally by Thiele [1895] and discussed, e.g., by Lackner and Skowron [2021] and Aziz *et al.* [2015]. Given a nondecreasing weight function $w \colon \mathbb{N} \to \mathbb{R}_+$ such that $w(0) = 0$, we define the w-Thiele score (w-score) of a committee $S = \{s_1, \dots, s_k\}$ in election E to be:

$$w\text{-score}_E(S) = \textstyle\sum_{v \in V} w(|A(v) \cap S|).$$

The w-Thiele rule outputs all committees with the highest w-score. We require that for each of our weight functions w, it is possible to compute each value $w(i)$ in polynomial time with respect to i. Additionally, we focus on functions such that $w(1) = 1$ and for each positive integer i it holds that $w(i) - w(i-1) \geq w(i+1) - w(i)$. We refer to such functions, and the Thiele rules that they define, as 1-concave. Three best-known 1-concave Thiele rules include the already defined AV rule, which uses function $w_{\text{AV}}(t) = t$, the approval-based Chamberlin–Courant rule (CCAV), which uses function $w_{\text{CCAV}}(t) = [t \geq 1]$, and the proportional approval voting rule (PAV), which uses the function $w_{\text{PAV}}(t) = \sum_{i=1}^{t} 1/i$.

While it is easy to compute some winning committees under the AV rule in polynomial time (out of possibly exponentially many), for the other Thiele rules, including CCAV and PAV, even deciding if a committee with at least a given score exists is NP-hard (see the works of Procaccia *et al.* [2008] and Betzler *et al.* [2013] for the case of CCAV, and the works of Aziz *et al.* [2015] and Skowron *et al.* [2016] for the general case). Hence, sometimes the following greedy variants of Thiele rules are used (E is the input election and k is the desired committee size):

> Let f be a w-Thiele rule. Its greedy variant, denoted Greedy-f, first sets $W_0 := \emptyset$ and then executes k iterations, where for each $i \in [k]$, in the i-th iteration it computes $W_i := W_{i-1} \cup \{c\}$ such that c is a candidate in $C \setminus W_{i-1}$ that maximizes the w-score of W_i. Finally, it outputs W_k. In case of internal ties, i.e., if at some iteration there is more than one candidate that the algorithm may choose, the algorithm outputs all committees that can be obtained for some way of resolving each of these ties. In other words, we use the parallel-universes tie-breaking model [Conitzer *et al.*, 2009].

When we discuss the operation of some Greedy-f rule on an election E and we discuss the situation after its i-th iteration, where, so far, subcommittee W_i was selected, then by the score of a (not-yet-selected) candidate c we mean the value $w\text{-score}_E(W_i \cup \{c\}) - w\text{-score}_E(W_i)$, i.e., the marginal increase of the w-score that would result from selecting c. We refer to the greedy variants of CCAV and PAV as GreedyCCAV and GreedyPAV (in the literature, these rules are also sometimes called *sequential* variants of CCAV and PAV, see, e.g., the book of Lackner and Skowron [2023]). Given a greedy variant of a 1-concave Thiele rule, it is always possible to compute at least one of its winning committees in polynomial time by breaking internal ties arbitrarily. Further, it is well-known that the w-score of this committee is at least a $1 - 1/e \approx 0.63$ fraction of the highest possible w-score; this follows from the classic result of Nemhauser *et al.* [1978] and the fact that w-score is monotone and submodular.

The Phragmén (sequential) rule proceeds as follows (see, e.g., the works of Janson [2016] and Brill *et al.* [2017]):

> Let $E = (C, V)$ be an election and let k be the committee size. Each candidate costs a unit of currency. The voters start with no money, but they receive it continuously at a constant rate. As soon as there is a group of voters who approve a certain not-yet-selected candidate and who together have a unit of currency, these voters "buy" this candidate (i.e., they give away all their money and the candidate is included in the committee). The process stops as soon as k candidates are selected. For internal ties, we use the parallel-universes tie-breaking.

Method of Equal Shares (MEqS), introduced by Peters and Skowron [2020] and Peters *et al.* [2021], is similar in spirit, but gives the voters their "money" up front (we use the same notation as above):

> Initially, each voter has a budget equal to $k/|V|$. The rule starts with an empty committee and executes up to k iterations as follows (for each voter v, let $b(v)$ denote v's budget in the current iteration): For each not-yet-selected candidate c we check if the voters that approve c have at least a unit of currency (i.e., $\sum_{v \in A(c)} b(v) \geq 1$). If so, then we compute value ρ_c such that $\sum_{v \in A(c)} \min(b(v), \rho_c) = 1$, which we call the per-voter cost of c. We extend the committee with this candidate c', whose per-voter cost $\rho_{c'}$ is the lowest; the voters approving c' "pay" for him or her (i.e., each voter $v \in A(c')$ gives away $\min(b(v), \rho_{c'})$ of his or her budget). In case of internal ties, we use the parallel-universes tie-breaking. The process stops as soon as no candidate can be selected.

The above process, referred to as Phase 1 of MEqS, often selects fewer than k candidates. To deal with this, we extend the committee with candidates selected by Phragmén (started off with the budgets that the voters had at the end of Phase 1). We jointly refer to the greedy rules, Phragmén, MEqS, and Phase 1 of MEqS as sequential rules.

We assume familiarity with basic classes of computational complexity such as P, NP, and coNP. #P is the class of functions that can be expressed as counting accepting paths of nondeterministic polynomial-time Turing machines. Additionally, we consider parameterized complexity classes such as FPT and W[1] (see, e.g., the textbooks of Niedermeier [2006] and Cygan *et al.* [2015]). #W[1] is a parameterized counting class which relates to W[1] in the same way as #P relates to NP [Flum and Grohe, 2004]. When discussing counting problems, it is standard to use Turing reductions: A counting problem $\#A$ reduces to a counting problem $\#B$ if there is a polynomial time algorithm that solves $\#A$ in polynomial time, provided it has oracle access to $\#B$ (i.e., it can solve $\#B$ in constant time).[2]

3 Unique Winning Committee

In this section we consider the problem of deciding if a given multiwinner rule outputs a unique committee in a given election. Formally, we are interested in the following problem.

Definition 3.1. *Let f be a multiwinner voting rule. In the f-UNIQUE-COMMITTEE problem we are given an election E and a committee size k, and we ask if $|f(E,k)| = 1$.*

It is a folk result that for AV and SAV this problem is in P (see beginning of Section 4 for an argument). For Thiele rules other than AV, the situation is more intriguing. In particular, already the problem of deciding if a given committee is winning under the CCAV rule is coNP-complete [Sonar *et al.*, 2020]. We show that for 1-concave Thiele rules other than AV the UNIQUE-COMMITTEE problem is coNP-hard (and we conjecture that the problem is not in coNP).

[2] For #W[1], the running time can even be larger, but our #W[1]-hardness proofs use polynomial-time reductions.

Proposition 3.1. *Let f be a 1-concave w-Thiele rule other than AV. Then f-UNIQUE-COMMITTEE is coNP-hard.*

Proof. Let $x = w(2) - w(1)$ and assume, for now, that $x < 1$. We give a reduction from INDEPENDENT-SET to the complement of f-UNIQUE-COMMITTEE. An instance of INDEPENDENT-SET consists of a graph G and an integer k, and we ask if there are k vertices neither of which is connected with the others. Let G' be a graph obtained from G by adding k vertices such that each of the new vertices is connected to each of the old ones (but the new vertices are not connected to each other). If G does not have a size-k independent set, then G' has a unique one, and if G has at least one size-k independent set, then G' has at least two. Let us denote the vertices of G' as $V(G') = \{v_1, \dots, v_n\}$ and its edges as $E(G') = \{e_1, \dots, e_m\}$. Let $d(v_i)$ be the degree of a vertex v_i and δ be the highest degree in $V(G')$. We fix the committee size to be k and we form an election E with the candidate set $V(G')$ and with the following voters:

1. For each edge $e_\ell = \{v_i, v_j\}$ there is a single voter who approves v_i and v_j.
2. For each vertex v_i there are $\delta - d(v_i)$ voters approving v_i.

Consider a set of k vertices from $V(G')$. If this set is an independent set, then interpreted as a committee in election E, it has w-score equal to δk. On the other hand, if S is not an independent set, then its score is at most $(\delta k - 1) + x < \delta k$. We know that G' has an independent set of size k. If G also has one, then our election has at least two winning committees and, otherwise, the winning committee is unique. The case where $x = 1$ is in the full version of the paper. $\square$

For greedy variants of Thiele rules (with the natural exception of AV) and for the Phragmén rule, deciding if the winning committee is unique is coNP-complete. Our proof for the greedy variants of Thiele rules is inspired by a complexity-of-robustness proof for GreedyPAV, provided by Faliszewski *et al.* [2022]. For Phragmén, somewhat surprisingly, their robustness proof directly implies our desired result. We also get analogous result for Method of Equal Shares and its Phase 1.

Theorem 3.2. *Let f be a 1-concave w-Thiele rule, $f \neq AV$. Greedy-f-UNIQUE-COMMITTEE is coNP-complete.*

Corollary 3.3. UNIQUE-COMMITTEE *is coNP-complete for GreedyCCAV, GreedyPAV, and* PHRAGMÉN.

Theorem 3.4. UNIQUE-COMMITTEE *is coNP-complete for Phase 1 of MEqS.*

Proof. One can verify that the problem is in coNP. To show hardness, we give a reduction from the complement of the classic NP-complete problem, X3C. An instance of X3C consists of a universe set $U = \{u_1, \dots, u_{3n}\}$ and a family $\mathcal{S} = \{S_1, \dots, S_{3n}\}$ of size-3 subsets of U. We ask if there are n sets from $\mathcal{S}$ whose union is U (we refer to such a family as an exact cover of U; note that the sets in such a cover must be disjoint). Without loss of generality, we assume that each member of U belongs to exactly three sets from $\mathcal{S}$ [Gonzalez, 1985] and that n is even.

Now we describe our election. Ideally, we would like to distribute different amounts of budget between different voters, but as MEqS splits the budget evenly, we design the election in such a way that in the initial iterations the respective voters spend appropriate amounts of money on the candidates that otherwise are not crucial for the construction. We form the following groups of voters (we reassure the reader that the analysis is more pleasant than it may appear):

1. Group B, which contains $144n^3 - 12n$ voters.
2. Group B_U, which contains $54n^3 + 9n^2$ voters.
3. Group U', which models the elements of the universe set U. For each $u_i \in U$, there is a single corresponding voter in U'. We have $|U'| = 3n$.
4. Group U'', which serves a similar purpose as U', but contains more voters. Specifically, for each $u_i \in U$, there are $6n$ corresponding voters in U''; $|U''| = 18n^2$.
5. Group V_{pd}, which contains $12n$ voters.
6. Group V_S, which contains $9n$ voters.
7. Two voters, d_1 and d_2.

In total, there are $198n^3 + 27n^2 + 12n + 2$ voters. Further, we have the following groups of candidates:

1. Group C_B of $144n^3 - 12n^2$ candidates approved by the $144n^3$ voters from $B \cup V_{pd}$.
2. Group C_U of $54n^3 + 24n^2 + {}^{5n}\!/\!{}_2$ candidates approved by the $54n^3 + 27n^2 + 3n$ voters from $B_U \cup U' \cup U''$.
3. Candidate p approved by the $12n$ voters from V_{pd}.
4. Candidate d approved by the $15n$ voters from $V_{pd} \cup U'$.
5. Candidates c_1 and c_2, both approved by d_1 and d_2.
6. Group D of $15n^2 + \frac{45n}{2} + 5$ candidates approved by d_1.
7. For each set $S_\ell \in \mathcal{S}$ such that $S_\ell = \{u_i, u_j, u_t\}$ we have a corresponding candidate s_ℓ approved by: (a) three unique voters from V_S, (b) the voters from U' and U'' that correspond to the elements u_i, u_j, u_t. We write S to denote this group of candidates and we refer to its members as the S-candidates. Each S-candidate is approved by $3 + 3 + 3 \cdot 6n = 18n + 6$ voters.

We have $198n^3+27n^2+28n+9$ candidates in total. We set the committee size k to be equal to the number of voters, i.e., $k = 198n^3 + 27n^2 + 12n + 2$. Let us consider the following two committees (note that each of them contains fewer than k candidates; indeed, Phase 1 of MEqS sometimes chooses committees smaller than requested):

$$W_d = C_B \cup C_U \cup S \cup \{c_1, c_2\} \cup \{d\},$$
$$W_p = C_B \cup C_U \cup S \cup \{c_1, c_2\} \cup \{p\}.$$

We claim that Phase 1 of MEqS always outputs committee W_d, and if $(U, \mathcal{S})$ is a *yes*-instance then it also outputs W_p.

Let us analyze how Phase 1 of MEqS proceeds on our election. Since the committee size is equal to the number of voters, initially each voter receives budget equal to 1.

At first, we will select all candidates from C_B. Indeed, there are $144n^3 - 12n^2$ candidates in this group, each approved by $144n^3$ voters (from $B \cup V_{pd}$). Each of these voters pays $1/144n^3$ for each of the candidates (this is the lowest per-voter candidate cost at this point). After these purchases, each voter from $B \cup V_{pd}$ will be left with budget equal to $1 - (144n^3 - 12n^2) \cdot (1/144n^3) = 1/12n$.

Next, we will select all candidates from C_U. Indeed, this set contains $54n^3 + 24n^2 + {}^{5n}\!/\!{}_2$ candidates approved by $54n^3 + 27n^2 + 3n$ voters (from $B_U \cup U' \cup U''$) who have not spent any part of their budget yet. All candidates in C_U will be purchased at the same pre-voter cost of $1/(54n^3+27n^2+3n)$ (the lowest one at this point). Each voter in $B_U \cup U' \cup U''$ will be left with budget equal to $1 - (54n^3 + 24n^2 + {}^{5n}\!/\!{}_2) \cdot 1/(54n^3+27n^2+3n) = \frac{3n^2+n/2}{54n^3+27n^2+3n} = \frac{6n+1}{108n^2+54n+6} = \frac{6n+1}{(6n+1)\cdot(18n+6)} = 1/(18n+6)$.

Next, we consider the S-candidates who, at this point, have the highest approval score among the yet unselected candidates. As each S-candidate is approved by exactly $18n + 6$ voters and each voter still has budget higher or equal to $1/(18n+6)$, we keep selecting the S-candidates at the per-voter cost of $1/(18n+6)$ as long as there is at least one such candidate whose all voters still have budget of at least $1/(18n+6)$.

Upon selecting a given S-candidate, corresponding to set S_ℓ, all the voters who approve him or her pay $1/(18n+6)$. This includes the three unique voters from V_S and the voters from U' and U'' who correspond to the members of S_ℓ. Prior to this payment, the voters from U' and U'' have budget equal to $1/(18n+6)$, so they end up with 0 afterward (and we say that they are *covered* by this S-candidate). Consequently, the S-candidates that we buy at the per-voter cost of $1/(18n+6)$ correspond to disjoint sets.

Now let us consider what happens when there is no S-candidate left who can be purchased at the per-voter cost of $1/(18n+6)$. This means that for each unselected S candidate, at least $6n + 1$ voters approving him have already been covered and have no budget left. Hence, for a given S-candidate there are at least $6n+1$ voters (from U' and U'') whose budget is 0, at most $12n+2$ voters (from U' and U'') who each have budget of $1/(18n+6)$, and three voters (from V_S) who each have budget equal to 1. To buy this S candidate, the voters from U' and U'' would have to use up their whole budget, and the voters from V_S would have to pay at least:

$$\tfrac{1}{3}(1 - (12n + 2) \cdot \tfrac{1}{18n+6}) = \tfrac{18n+6-(12n+2)}{3\cdot(18n+6)} = \tfrac{3n+2}{27n+9}$$

each. However, at this point there are two candidates that can be purchased at lower per-voter cost.

Indeed, candidate p could be purchased by the $12n$ voters from V_{pd} at the per-voter cost of $1/12n$ (after buying the candidates from C_B, they still have exactly this amount of budget left). Since candidate d also is approved by all the voters from V_{pd}, and also by the voters from U', candidate d would either have the same per-voter cost as p (in case all the members of U' were already covered) or would have an even lower per-voter cost. The only other remaining candidates are c_1, c_2, and the candidates from D, but their per-voter costs are greater or equal to $1/2$. Hence, at this point, MEqS either selects p or d. The former is possible exactly if the already selected S-candidates form an exact cover of U' (and, hence, correspond to an exact cover for our input instance of X3C).

If we select p, then the $12n$ voters from V_{pd} use up all their budget. The remaining voters who approve d, those in

U', have total budget equal to at most $3n \cdot \frac{1}{18n+6} < 1$, so d cannot be selected in any of the following iterations (within Phase 1). On the other hand, if we select d, then all the voters from U' would have to pay all they had left (that is, either 0 or $\frac{1}{18n+6}$, each) and voters from V_{pd} would split the remaining cost. That is, each voter from V_{pd} would have to pay at least:

$$\frac{1-3n\cdot\frac{1}{18n+6}}{12n} = \frac{18n+6-3n}{12n\cdot(18n+6)} = \frac{15n+6}{12n\cdot(18n+6)}.$$

Consequently, each voter from V_{pd} would be left with at most:

$$\frac{1}{12n} - \frac{15n+6}{12n\cdot(18n+6)} = \frac{18n+6-(15n+6)}{12n\cdot(18n+6)} = \frac{1}{72n+24}.$$

This would not suffice to purchase p, as $12n \cdot \frac{1}{72n+24} < 1$. Thus either we select d (and not p) or we select p (and not d; where this is possible only if we previously purchased S-candidates that cover all members of U').

In the following iterations, we purchase all remaining S-candidates (because each of them is approved by three unique voters from V_S), as well as candidates c_1 and c_2 (voters d_1 and d_2 buy them with per-voter cost of $1/2$ for each). This uses up the budget of d_1 and, so, no candidate from D is selected. All in all, if there is no exact cover for the input X3C instance, then W_d is the unique winning committee, but otherwise W_d and W_p tie. This finishes the proof. □

UNIQUE-COMMITTEE is also coNP-complete for the full version of MEqS. To see this, it suffices to note that after adding enough voters with empty votes, MEqS becomes equivalent to Phragmén (because per-voter budget is so low that Phase 1 becomes vacuous) and inherits its hardness.

On the positive side, for sequential rules we can solve UNIQUE-COMMITTEE in FPT time with respect to the committee size: In essence, we first compute some winning committee and then we try all ways of breaking internal ties to find a different one. For small values of k, such as, e.g., $k \leq 10$, the algorithm is fast enough to be practical.

Theorem 3.5. *Let f be MEqS, Phase 1 of MEqS, Phragmén, or a greedy variant of a 1-concave Thiele rule. There is an FPT algorithm for f-UNIQUE-COMMITTEE parameterized by the committee size.*

Proof. Let E be the input election and let k be the committee size. First, we compute some committee W in $f(E,k)$, by running the algorithm for f and breaking the internal ties arbitrarily. Next, we rerun the algorithm, but whenever it is about to add a candidate into the constructed committee, we do as follows (let T be the set of candidates that the algorithm can insert into the committee): If T contains some candidate c that does not belong to W, then we halt and indicate that there are at least two winning committees (W and those that include c). If T is a subset of W, then we recursively try each way of breaking the tie. If the algorithm completes without halting, we report that there is a unique winning committee (see the full version of the paper for correctness argument and running time analysis). □

For 1-concave Thiele rules other than AV, UNIQUE-COMMITTEE is co-W[1]-hard when parameterized by the committee size (this follows from the proof of Proposition 3.1 as INDEPENDENT-SET is W[1]-hard for parameter k). To solve the problem in practice, we note that for each 1-concave Thiele rule there is an integer linear program (ILP) whose solution corresponds to a winning committee. We can either use the ability of some ILP solvers to output several solutions (which only succeeds in case of a tie), or we can use the following strategy: First, we compute some winning committee using the basic ILP formulation. Then, we extend the formulation with a constraint that requires the committee to be different from the previous one and compute a new one. If both committees have the same score, then there is a tie.

4 Counting Winning Committees

Let us now consider the problem of counting the winning committees. Formally, our problem is as follows.

Definition 4.1. *Let f be a multiwinner voting rule. In the f-#WINNING-COMMITTEES problem we are given an election and a committee size k; we ask for $|f(E,k)|$.*

There are polynomial-time algorithms for computing the number of winning committees for AV and SAV. For an election E with committee size k, we first sort the candidates with respect to their scores in a non-increasing order and we let x be the score of the k-th candidate. Then, we let S be the number of candidates whose score is greater than x, and we let T be the number of candidates with score equal to x. There are $\binom{T}{k-S}$ winning committees.

Proposition 4.1. *{AV, SAV}-#WINNING-COMMITTEES* $\in$ P

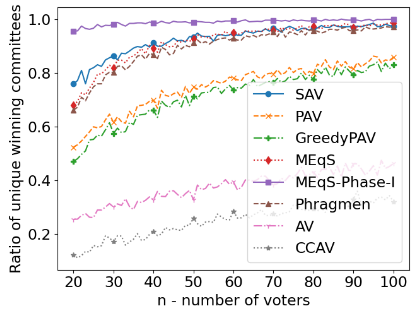
On the other hand, whenever f-UNIQUE-COMMITTEE is intractable, so is f-#WINNING-COMMITTEES. Indeed, it immediately follows that there is no polynomial-time $(2-\varepsilon)$-approximation algorithm for f-#WINNING-COMMITTEES for any $\varepsilon > 0$ (such an algorithm could solve f-UNIQUE-COMMITTEE in polynomial time as for an election with a single winning committee it would have to output 1, and for an election with more winning committees it would have to output an integer greater or equal $\frac{2}{2-\varepsilon} > 1$, so we could distinguish these cases[3]). Yet, we have a stronger result.

Proposition 4.2. *Let f be a 1-concave Thiele rule (different from AV), its greedy variant, Phragmén, MEqS or Phase 1 of MEqS. Unless P = NP, there is no polynomial-time approximation algorithm for f-#WINNING-COMMITTEES with polynomially-bounded approximation ratio.*[4]

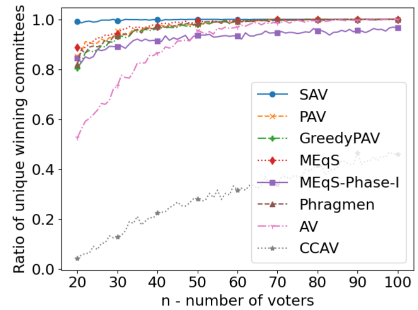
We note that the construction given in the proof of Proposition 3.1 also shows that for each 1-concave Thiele rule $f \neq$ AV, f-#WINNING-COMMITTEES is both #P-hard and #W[1]-hard for parameterization by the committee size (this reduction produces elections that have one more winning committees than the number of size-k independent sets in the input graph, and counting independent sets is

[3]We assume here that if a solution for a counting problem is $x \in \mathbb{N}$, then an α-approximation algorithm, with $\alpha \geq 1$, has to output an integer between x/α and αx. If we allowed rational values on output, the inapproximability bound would drop to $\sqrt{2} - \varepsilon$.

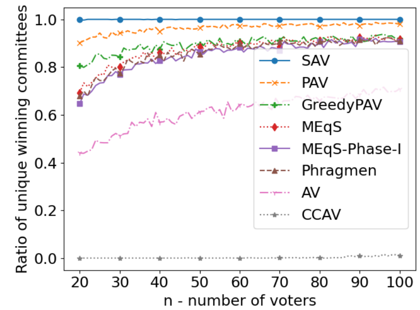
[4]There is no polynomial p such that there exists a p-approximation algorithm solving f-#WINNING-COMMITTEES



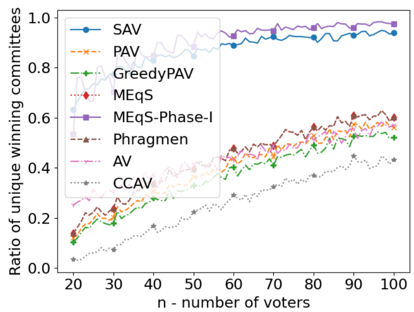
(a) $m = 30$, $k = 5$, $k/2$ approvals/vote, resampling model, $\phi = 0.75$



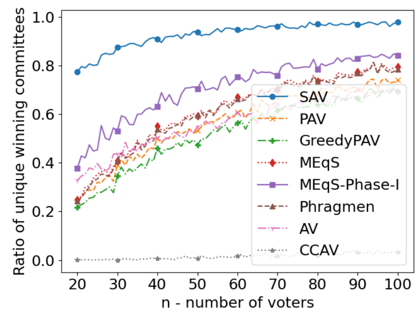
(b) $m = 30$, $k = 5$, k approvals/vote resampling model, $\phi = 0.75$



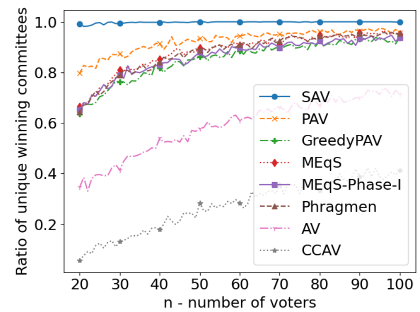
(c) $m = 30$, $k = 5$, $2k$ approvals/vote resampling model, $\phi = 0.75$



(d) $m = 30$, $k = 5$, $k/2$ approvals/vote Interval



(e) $m = 30$, $k = 5$, k approvals/vote Interval



(f) $m = 30$, $k = 5$, PabuLib (Warsaw)

Figure 1: Results of our experiments. By "$k/2$ approvals/vote" we mean that on average a single vote contains approximately $k/2$ approvals (the meaning of k and $2k$ is analogous).

both #P-complete and #W[1]-complete for parameterization by k [Valiant, 1979; Flum and Grohe, 2004]). For greedy variants of 1-concave Thiele rules and Phragmén, we give a new hardness proof, as UNIQUE-COMMITTEE is in FPT.

Theorem 4.3. *Let f be Phragmén or a greedy variant of a 1-concave Thiele rule (different from AV). f-#WINNING-COMMITTEES is #P-hard and #W[1]-hard (for the parameterization by the committee size).*

Proof. We first consider greedy variants of 1-concave Thiele rules. Let w be the weight function used by f. Let $x = w(2) - w(1)$. We have $w(1) = 1$ and we assume that $x < 1$ (we will consider the other case later). We show a reduction from the #MATCHING problem, where we are given a graph G, an integer k, and we ask for the number of size-k matchings (i.e., the number of size-k sets of edges such that no two edges in the set share a vertex). #MATCHING is #P-complete and #W[1]-hard for parameterization by k [Curticapean and Marx, 2014].

Let G and k be our input. We form an election E where the edges of G are the candidates and the vertices are the voters. For each edge $e = \{u, v\}$, the corresponding edge candidate is approved by the vertex voters corresponding to u and v. We also form an election E_p, equal to E except that it has two extra voters who both approve a single new candidate, p.

We note that every candidate in both E and E_p is approved by exactly two voters. Hence, the greedy procedure first keeps on choosing candidates whose score is 2 (i.e., edges that jointly form a matching, or candidate p in E_p). It selects the candidates with lower scores (i.e., edges that break a matching) only when score-2 candidates disappear.

Let W be some size-k f-winning committee for election E_p. We consider two cases:

1. If p does not belong to W, then the edge candidates in W form a matching. If it were not the case, then before including an edge candidate with score lower than 2, the greedy algorithm would include p in the committee.
2. If p belongs to W then $W \setminus \{p\}$ is an f-winning committee of size $k - 1$ for election E. Indeed, if we take the run of the greedy algorithm that computes W and remove the iteration where p is selected, we get a correct run of the algorithm for election E and committee size $k - 1$. Further, for every size-$(k - 1)$ committee winning in E, $S \cup \{p\}$ is a size-k winning committee in E_p (because we can always select p in the first iteration).

So, to compute the number of size-k matchings in G, we take the number of winning size-k committees in E_p and subtract from it the number of winning size-$(k - 1)$ committees in E. Rest of the argument is in the full paper. □

Corollary 4.4. *#WINNING-COMMITTEES is #P-hard and #W[1]-hard (for the parameterization by the committee size)* *for GreedyCCAV, GreedyPAV, Phragmén, and MEqS.*

The above result holds for MEqS because of its relation to Phragmén. For Phase 1 of MEqS, we have $\#\mathrm{P}$-hardness but $\#\mathrm{W}[1]$-hardness so far remains elusive.

Theorem 4.5. $\#$WINNING-COMMITTEES *is* $\#\mathrm{P}$*-hard for Phase 1 of MEqS.*

5 Experiments

A'priori, it is not clear how frequent ties are in multiwinner elections. In this section we present experiments that show that they are quite common, at least if one considers elections of moderate size. Our code is available at https://github.com/Project-PRAGMA/Ties-IJCAI-2023.

5.1 Statistical Cultures and the Basic Experiment

Below we describe how we generate elections and how we perform our basic experiments.

Resampling Model [Szufa *et al.*, 2022] We have two parameters, p and ϕ, both between 0 and 1. To generate an election with candidate set $C = \{c_1, \dots, c_m\}$ and with n voters, we first choose uniformly at random a central vote u approving exactly $\lfloor pm \rfloor$ candidates. Then, we generate the votes, for each considering the candidates independently, one by one. For a vote v and candidate c, with probability $1 - \phi$ we copy c's approval status from u to v (i.e., if u approves c, then so does v; if u does not approve c then neither does v), and with probability ϕ we "resample" the approval status of c, i.e., we let v approve c with probability p. On average, each voter approves about pm candidates.

Interval Model. In the Interval model, each voter and each candidate is a point on the interval $[0, 1]$, chosen uniformly at random. Additionally, each candidate c has radius r_c and a voter v approves candidate c if the distance between their points is at most r_c. Intuitively, the larger the radius, the more appealing a given candidate is. We generate the radii of the candidates by taking a base radius r as input and, then, choosing each candidates' radius from the normal distribution with mean r and standard deviation $r/2$. Such spatial models are discussed in detail, e.g., by Enelow and Hinich; Enelow and Hinich [1984; 1990]. In the approval setting, they were recently considered, e.g., by Bredereck *et al.* [2019] and Godziszewski *et al.* [2021].

PabuLib Data. PabuLib is a library of real-life participatory budgeting (PB) instances, mostly from Polish cities [Faliszewski *et al.*, 2023]. A PB instance is a multiwinner election where the candidates (referred to as projects) have costs and the goal is to choose a "committee" of at most a given total cost. We restrict our attention to instances from Warsaw, which use approval voting, and we disregard the cost information (while this makes our data less realistic, we are not aware of other sources of real-life data for approval elections that would include sufficiently large candidate and voter sets). To generate an election with m candidates and n voters, we randomly select a Warsaw PB instance, remove all but m candidates with the highest approval score, and randomly draw n voters (with repetition, restricting our attention only to voters who approve at least one of the remaining candidates). We also considered several other strategies of choosing the candidates and obtained very similar results. Another argument for focusing on the most approved candidates is that the voters cared about them the most (in aggregate). We consider 120 PB instances from Warsaw that include at least 30 candidates (and each of them includes at least a thousand votes).

Basic Experiment. In a basic experiment we fix the number of candidates m, the committee size k, and a statistical culture. Then, for each number n of voters between 20 and 100 (with a step of 1) we generate 1000 elections with m candidates and n voters, and for each of them compute whether our rules have a unique winning committee (we omit Greedy-CCAV). Then we present a figure that on the x axis has the number of voters and on the y axis has the fraction of elections that had a unique winning committee for a given rule. For AV and SAV, we use the algorithm from the beginning of Section 4, for sequential rules we use the FPT algorithm from Theorem 3.5, and for CCAV and PAV we use the ILP-based approach, with a solver that provides multiple solutions.

5.2 Results

All our experiments regard 30 candidates and committee size 5. First, we performed three basic experiments for the resampling model with the parameter p (approval probability) set so that, on average, each voter approved either $k/2$, k, or $2k$ candidates. We used $\phi = 0.75$ (according to the results of Szufa *et al.* [2022], this value gives elections that resemble the real-life ones). We present the results in the top row of Figure 1. Next, we also performed two basic experiments for the Interval model (with the base radius selected so that, on average, each voter approved either $k/2$ or k candidates), and with the PabuLib data (see the second row of Figure 1). These experiments support the following general conclusions.

First, for most scenarios and for most of our rules, there is a nonnegligible probability of having a tie (where, depending on the rule and the number of voters, this probability may be as low as 5% or as high as nearly 100%). This justifies why one needs to be ready to detect and handle ties in moderately sized multiwinner elections.

Second, we see that SAV generally leads to fewest ties, CCAV leads to most, and AV often holds a strong second position in this category. The other rules are in between. Phase 1 of MEqS often has significantly fewer ties than the other rules, but full version of MEqS does not stand out. PAV occasionally leads to fewer ties (in particular, on PabuLib data and on the resampling model with $2k$ approvals per vote).

6 Summary

We have shown that, in general, detecting ties in multiwinner elections is intractable, but doing so for moderately-sized ones is perfectly possible. Our experiments show that ties in such elections are a realistic possibility and one should be ready to handle them. Intractability of counting winning committees suggests that tie-breaking by sampling committees may not be feasible. Looking for fair tie-breaking mechanisms is a natural follow-up research direction.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 101002854).

References

[Aziz *et al.*, 2015] H. Aziz, S. Gaspers, J. Gudmundsson, S. Mackenzie, N. Mattei, and T. Walsh. Computational aspects of multi-winner approval voting. In *Proceedings of AAMAS-2015*, pages 107–115, 2015.

[Betzler *et al.*, 2013] N. Betzler, A. Slinko, and J. Uhlmann. On the computation of fully proportional representation. *Journal of Artificial Intelligence Research*, 47:475–519, 2013.

[Bredereck *et al.*, 2019] R. Bredereck, P. Faliszewski, A. Kaczmarczyk, and R. Niedermeier. An experimental view on committees providing justified representation. In *Proceedings of IJCAI-2019*, pages 109–115, 2019.

[Brill *et al.*, 2017] M. Brill, R. Freeman, S. Janson, and M. Lackner. Phragmén's voting methods and justified representation. In *Proceedings of AAAI-2017*, pages 406–413, 2017.

[Conitzer *et al.*, 2009] V. Conitzer, M. Rognlie, and L. Xia. Preference functions that score rankings and maximum likelihood estimation. In *Proceedings of IJCAI-2009*, pages 109–115. AAAI Press, July 2009.

[Curticapean and Marx, 2014] R. Curticapean and D. Marx. Complexity of counting subgraphs: Only the boundedness of the vertex-cover number counts. In *Proceedings of FOCS-2014*, pages 130–139, 2014.

[Cygan *et al.*, 2015] M. Cygan, F. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.

[Elkind *et al.*, 2017] E. Elkind, P. Faliszewski, P. Skowron, and A. Slinko. Properties of multiwinner voting rules. *Social Choice and Welfare*, 48(3):599–632, 2017.

[Enelow and Hinich, 1984] J. Enelow and M. Hinich. *The Spatial Theory of Voting: An Introduction*. Cambridge University Press, 1984.

[Enelow and Hinich, 1990] J. Enelow and M. Hinich. *Advances in the Spatial Theory of Voting*. Cambridge University Press, 1990.

[Faliszewski *et al.*, 2017] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Multiwinner voting: A new challenge for social choice theory. In U. Endriss, editor, *Trends in Computational Social Choice*. AI Access Foundation, 2017.

[Faliszewski *et al.*, 2022] P. Faliszewski, G. Gawron, and B. Kusek. Robustness of greedy approval rules. In *Proceedings of EUMAS-2022*, pages 116–133, 2022.

[Faliszewski *et al.*, 2023] P. Faliszewski, J. Flis, D. Peters, G. Pierczyński, P. Skowron, D. Stolicki, S. Szufa, and N. Talmon. Participatory budgeting: Data, tools and analysis. In *Proceedings of IJCAI-2023*, 2023.

[Flum and Grohe, 2004] J. Flum and M. Grohe. The parameterized complexity of counting problems. *SIAM Journal on Computing*, 33(4):892–922, 2004.

[Freeman *et al.*, 2015] R. Freeman, M. Brill, and V. Conitzer. General tiebreaking schemes for computational social choice. In *Proceedings of AAMAS-2015*, pages 1401–1409, 2015.

[Godziszewski *et al.*, 2021] M. Godziszewski, P. Batko, P. Skowron, and P. Faliszewski. An analysis of approval-based committee rules for 2D-Euclidean elections. In *Proceedings of AAAI-2021*, pages 5448–5455, 2021.

[Gonzalez, 1985] T. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:293–306, 1985.

[Janeczko and Faliszewski, 2023] L. Janeczko and P. Faliszewski. Ties in multiwinner approval voting. Technical Report arXiv:2305.01769 [cs.GT], arXiv.org, May 2023.

[Janson, 2016] S. Janson. Phragmén's and Thiele's election methods. Technical Report arXiv:1611.08826 [math.HO], arXiv.org, November 2016.

[Lackner and Skowron, 2021] M. Lackner and P. Skowron. Consistent approval-based multi-winner rules. *Journal of Economic Theory*, 192:105173, 2021.

[Lackner and Skowron, 2023] M. Lackner and P. Skowron. *Multi-Winner Voting with Approval Preferences*. Springer, 2023.

[Nemhauser *et al.*, 1978] G. Nemhauser, L. Wolsey, and M. Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming*, 14(1):265–294, 1978.

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

[Obraztsova and Elkind, 2011] S. Obraztsova and E. Elkind. On the complexity of voting manipulation under randomized tie-breaking. In *Proceedings of IJCAI-2011*, pages 319–324, 2011.

[Obraztsova *et al.*, 2011] S. Obraztsova, E. Elkind, and N. Hazon. Ties matter: Complexity of voting manipulation revisited. In *Proceedings of AAMAS-2011*, pages 71–78, 2011.

[Peters and Skowron, 2020] D. Peters and P. Skowron. Proportionality and the limits of welfarism. In *Proceedings of EC-2020*, pages 793–794, 2020.

[Peters *et al.*, 2021] D. Peters, G. Pierczyński, and P. Skowron. Proportional participatory budgeting with additive utilities. In *Proceedings of NeurIPS-2021*, pages 12726–12737, 2021.

[Procaccia *et al.*, 2008] A. Procaccia, J. Rosenschein, and A. Zohar. On the complexity of achieving proportional representation. *Social Choice and Welfare*, 30(3):353–362, 2008.

[Skowron *et al.*, 2016] P. Skowron, P. Faliszewski, and J. Lang. Finding a collective set of items: From proportional multirepresentation to group recommendation. *Artificial Intelligence*, 241:191–216, 2016.

[Sonar *et al.*, 2020] C. Sonar, P. Dey, and N. Misra. On the complexity of winner verification and candidate winner for multiwinner voting rules. In *Proceedings of IJCAI-2020*, pages 89–95, 2020.

[Szufa *et al.*, 2022] S. Szufa, P. Faliszewski, L. Janeczko, M. Lackner, A. Slinko, K. Sornat, and N. Talmon. How to sample approval elections? In *Proceedings of IJCAI-2022*, pages 496–502, 2022.

[Thiele, 1895] T. Thiele. Om flerfoldsvalg. In *Oversigt over det Kongelige Danske Videnskabernes Selskabs Forhandlinger*, pages 415–441. 1895.

[Valiant, 1979] L. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189–201, 1979.

[Xia, 2021] L. Xia. How likely are large elections tied? In *Proceedings of EC-2021*, pages 884–885, 2021.

[Xia, 2022] L. Xia. Fair and fast tie-breaking for voting. Technical Report arXiv:2204.14838 [cs.GT], arXiv.org, May 2022.