Computational Complexity of Two Variants of the Possible Winner Problem

Dorothea Baumeister Magnus Roos Jörg Rothe
Institut für Informatik
Heinrich-Heine-Universität Düsseldorf
40225 Düsseldorf, Germany
{baumeister, roos, rothe}@cs.uni-duesseldorf.de

ABSTRACT

A possible winner of an election is a candidate that has, in some kind of incomplete-information election, the possibility to win in a complete extension of the election. The first type of problem we study is the Possible co-Winner with respect to the Addition of New Candidates (PcWNA) problem, which asks, given an election with strict preferences over the candidates, is it possible to make a designated candidate win the election by adding a limited number of new candidates to the election? In the case of unweighted voters we show NP-completeness of PcWNA for a broad class of pure scoring rules. We will also briefly study the case of weighted voters. The second type of possible winner problem we study is Possible Winner/co-Winner under Uncertain Voting System (PWUVS and PcWUVS). Here, uncertainty is present not in the votes but in the election rule itself. For example, PcWUVS is the problem of whether, given a set C of candidates, a list of votes over C, a distinguished candidate $c \in C$, and a class of election rules, there is at least one election rule from this class under which c wins the election. We study these two problems for a class of systems based on approval voting, the family of Copeland$^\alpha$ elections, and a certain class of scoring rules. Our main result is that it is NP-complete to determine whether there is a scoring vector that makes c win the election, if we restrict the set of possible scoring vectors for an m-candidate election to those of the form $(\alpha_1, \ldots, \alpha_{m-4}, x_1, x_2, x_3, 0)$, with $x_i = 1$ for at least one $i \in \{1, 2, 3\}$.

Categories and Subject Descriptors

F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity;
I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent Systems*;
J.4 [**Computer Applications**]: Social and Behavioral Sciences—*Economics*

General Terms

Economics, Theory

Keywords

voting protocols, social choice theory, computational social choice, possible winner problem

Cite as: Computational Complexity of Two Variants of the Possible Winner Problem, Dorothea Baumeister, Magnus Roos, and Jörg Rothe, *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, Tumer, Yolum, Sonenberg and Stone (eds.), May, 2–6, 2011, Taipei, Taiwan, pp. 853-860.

1. INTRODUCTION

A central task in computational social choice is the study of the algorithmic and computational properties of voting systems (see, e.g., the bookchapters [15, 4]). One of the classical problems in this field is the Manipulation problem, which deals with the question of whether a voter can benefit from strategic behavior. The celebrated Gibbard–Satterthwaite theorem [18, 23] says that in every nondictatorial voting system a strategic voter can alter the outcome of an election to his or her advantage by voting insincerely. Bartholdi et al. [2, 1] were the first to show that computational complexity can be used as a barrier to protect elections from manipulation attempts: In some voting systems, though manipulable in principle, it is computationally hard to compute successful manipulative preferences to cast.

Conitzer, Sandholm, and Lang [11] defined a more general version of this problem, called Coalitional Weighted Manipulation, where voters have weights and a whole group of voters can coordinate their strategic efforts. The complexity of this problem has been studied for many voting systems, including plurality, Borda, veto, Copeland, STV, maximin, plurality with run-off, regular cup, randomized cup, and including a dichotomy result for the class of pure scoring rules [11, 19]. In the case of unweighted voters the complexity of coalitional manipulation is still unknown for most pure scoring rules.

Another generalization of Manipulation is the Possible Winner (PW) problem, which was first introduced by Konczak and Lang [21]. Here the voters do not provide linear orders over the candidates, but partial orders. The question is whether there is an extension of the partial orders into linear ones such that a distinguished candidate wins the election. Manipulation is the special case of PW in which all voters but one report linear orders and one voter reports no preference at all. This implies that NP-hardness results for the Manipulation problem carry over to the PW problem. For the important class of pure scoring rules and the case of unweighted voters, the computational complexity of this problem is also settled by a full dichotomy result (see [6, 5]): It is solvable in polynomial time for plurality and veto, and NP-complete for all other pure scoring rules. These results also hold for PcW, the corresponding co-winner problem.

One variant of the PW problem was defined by Chevaleyre et al. [9] and also studied by Xia et al. [25]: Possible co-Winner with respect to the Addition of New Candidates (PcWNA). In this setting the voters report linear orders over an initial set of candidates and after reporting their preferences some new candidates are introduced. The problem is to determine whether one distinguished candidate among the initial ones can be a winner if the voters' preferences are extended to linear orders over the initial and the new candidates. PcWNA is a special case of PcW and is in some sense dual to the coalitional manipulation problem [9, 25]. In particular, the NP-hardness results for the PcW problem are not inherited by PcWNA. Note that PcWNA is also closely related to—but different from—the problem of control via adding candidates [3, 20] and to the cloning problem in elections [13].

We study the problem PcWNA in the case of unweighted voters and pure scoring rules, giving a deeper insight into a question raised by Chevaleyre et al. [9]. They showed that if one new candidate is added in the case of unweighted voters, PcWNA is polynomial-time solvable for a certain class of pure scoring rules but is NP-complete for one specific pure scoring rule (see Table 1), and they asked if that result can be extended to other pure scoring rules. Our main result in Section 3 establishes NP-completeness of PcWNA for a whole class of pure scoring rules if one new candidate is added. This result is obtained even for the case of unweighted voters. In addition, we briefly study the complexity of the PcWNA problem in the case of weighted voters.

In the second setting we consider, the possible winner problem is related to uncertainty about the election rule used. A similar setting has been previously studied by several authors. Conitzer, Sandholm, and Lang [11] showed that the computational complexity of manipulation can be increased by using a random instantiation for the cup protocol. Pini et al. [22] studied the problem of determining winners by sequential majority voting if preferences may be incomplete and the agenda is uncertain.

In general, we study the problem Possible Winner/co-Winner under Uncertain Voting System (PWUVS and PcWUVS), which asks whether a distinguished candidate, after all votes have been cast, can be made a winner of the election by choosing one election rule from a given class of rules. Specifically we will consider this problem with respect to a class of systems based on approval voting, the family of Copeland$^\alpha$ elections [14], and a certain class of scoring rules. Walsh [24] proposed to investigate PWUVS for the class of scoring rules, but to the best of our knowledge this issue has not been studied before. As a main result in Section 4, we show that PcWUVS and PWUVS are NP-hard for scoring rules if we restrict the set of possible scoring vectors for an m-candidate election, $m \geq 4$, to those of the form $(\alpha_1, \ldots, \alpha_{m-4}, x_1, x_2, x_3, 0)$, with $x_i = 1$ for at least one $i \in \{1, 2, 3\}$. Note that some important scoring rules, such as Borda and veto for $m \geq 4$ candidates, are contained in this restricted set of scoring vectors.

A motivation for uncertainty about the voting system used is that this may prevent the voters from attempting to manipulate the election, since reporting an insincere preference might result in a worse outcome for them. For example, consider an election with three candidates (a, b, and c), nine sincere voters (six cast the vote $c > a > b$, two $b > a > c$, and one $b > c > a$), and three strategic voters (whose true preferences are $a > b > c$). If the strategic voters would know for sure that the election is held under the plurality rule (which values a first position by one point and all other positions by zero points), they might have an incentive to not waste their votes by voting sincerely ($a > b > c$) but rather to help their second preferred candidate, b, to tie for winner with c by casting the three votes $b > a > c$. However, if the election is held under the Borda rule (which, for three candidates, values a first position by two points, a second position by one point, and a last position by zero points), casting the three insincere votes $b > a > c$ would make their most despised candidate c win with 13 points in total (leaving b second with 12 points and a last with 11 points), whereas the three sincere votes $a > b > c$ would make their favorite candidate a win with 14 points in total (leaving c second with 13 points and b last with 9 points). This means that uncertainty about the scoring rule may give the voters a strong incentive to reveal their true preferences.

2. DEFINITIONS AND NOTATION

An election (C, V) is given by a set C of candidates and a list V of votes over C. In preference-based voting systems, each vote in V is a (strict) linear ordering of the candidates in C, where the underlying binary relation $>$ on C is *total* (either $c > d$ or $d > c$ for all $c, d \in C$, $c \neq d$), *transitive* (for all $c, d, e \in C$, if $c > d$ and $d > e$ then $c > e$), and *asymmetric* (for all $c, d \in C$, if $c > d$ then $d > c$ does not hold). Here, $c > d$ means that candidate c is (strictly) preferred to candidate d. A voting system is a rule to determine the winner(s) of an election. We will consider three different types of voting systems: (pure) scoring rules, Copeland$^\alpha$ elections, and (variants of) approval voting.

Scoring rules (a.k.a. scoring protocols): Each scoring rule with m candidates is specified by an m-dimensional scoring vector $\vec{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_m)$ satisfying that

$$\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_m, \tag{1}$$

where each weight α_j is a nonnegative integer. For an election (C, V), a candidate $c \in C$ ranked at jth position in a vote $v \in V$ receives α_j points from v. The *score of c in* (C, V), denoted by $score_{(C,V)}(c)$, is the sum of all points c receives from all voters in V, and the winners of (C, V) are the candidates with maximum score. We may assume that the last weight, α_m, in the scoring vector is always zero, since each scoring rule not satisfying this condition can easily be transformed into one that satisfies it (see [19]). Adopting a notion introduced by Betzler and Dorn [6], we say a scoring rule is *pure* if for each $m \geq 2$, the scoring vector for m candidates can be obtained from the scoring vector for $m-1$ candidates by inserting one additional weight at any position subject to satisfying (1). One class of pure scoring rules is *k-approval*. Here the scoring vector has a one in the first k positions and a zero in all remaining positions. 1-approval—which may be better known under the name *plurality*—has the vector $(1, 0, \ldots, 0)$, and $(m-1)$-approval for m candidates—which may be better known under the name *antiplurality* or *veto*—has the vector $(1, \ldots, 1, 0)$. Another prominent scoring rule is the Borda rule, which has the scoring vector $(m-1, m-2, \ldots, 1, 0)$ for m candidates.

Copeland$^\alpha$, for a rational number α, $0 \leq \alpha \leq 1$: The winners are determined by pairwise comparisons of the candidates. For each $c \in C$, let $win(c)$ denote the number of candidates c beats in a pairwise comparison, and let $tie(c)$ denote the number of candidates c ties with in a pairwise comparison. The *Copeland$^\alpha$ score of a candidate c* is $win(c) + \alpha \cdot tie(c)$, and the candidates with maximum score win the election.

Approval voting: Every voter either approves or disapproves of each candidate, and the *approval score of a candi-*

date is the sum of his or her approvals. The candidates with the highest approval score win the election.

In the above voting systems, if there is only one candidate with maximum score, he or she is the unique winner.

The Possible co-Winner with respect to the Addition of New Candidates problem for a given voting system $\mathcal{E}$ is defined as follows:

Name: $\mathcal{E}$-Possible co-Winner with respect to the Addition of New Candidates ($\mathcal{E}$-PcWNA).
Given: A set of candidates $C = \{c_1, \ldots, c_m\}$, a list of votes $V = \{v_1, \ldots, v_n\}$ that are linear orders over C, a set C' with $|C'| = k$, $k \in \mathbb{N}$, of new candidates, and a distinguished candidate $c \in C$.
Question: Is there an extension of the votes in V to linear orders over $C \cup C'$ such that c is a winner of the election held under voting system $\mathcal{E}$.

In contrast to the above-defined problem where uncertainty is in the preferences, in Section 4.1 we will study another possible winner problem where uncertainty is in the voting system itself. This is the Possible co-Winner under Uncertain Voting System problem for a given class $\mathcal{V}$ of voting systems, which formally is defined as follows:

Name: $\mathcal{V}$-Possible co-Winner under Uncertain Voting System (PcWUVS).
Given: An election $E = (C, V)$, with the set of candidates C, a list of voters V consisting of linear orders over C, and a distinguished candidate $c \in C$.
Question: Is there a voting system $\mathcal{E}$ in $\mathcal{V}$ such that c is a winner of the election held under $\mathcal{E}$?

The problem is stated for the co-winner case. The unique-winner variant, PWUVS, is defined analogously by replacing "a winner" by "the unique winner" in the Question field above.

For the study of the computational complexity of the problems defined above, we will always assume that voters are unweighted and that the number of both voters and candidates is unbounded, unless stated otherwise.

3. POSSIBLE WINNER WRT. THE ADDITION OF NEW CANDIDATES

3.1 Unweighted Voters

In this section we study the problem PcWNA for pure scoring rules in the case of unweighted voters. Table 1 shows the results about the complexity of PcWNA for pure scoring rules that are already known from earlier work [9, 10, 25], where it is always assumed that voters are unweighted and that the number of initial candidates is unbounded. In particular, PcWNA is in P for the Borda rule for any fixed number of candidates, yet is NP-complete for the scoring vector $(3, 2, 1, 0, \ldots, 0)$ when the number of candidates is unbounded. Thus, this NP-completeness result is about a more general problem and does not contradict with the polynomial-time solvability of Borda in the restricted case of four candidates.

We now extend the result of Chevaleyre et al. [9] that PcWNA is NP-complete for pure scoring rules with vector $(3, 2, 1, 0, \ldots, 0)$ when one new candidate is added by showing that NP-completeness of PcWNA holds even for the class of pure scoring rules of the form $(\alpha_1, \alpha_2, 1, 0, \ldots, 0)$ with $\alpha_1 > \alpha_2 > 1$.

Scoring rule	PcWNA
Plurality	in P (see [9])
Veto	in P (see [9])
Borda	in P (see [9])
2-Approval	in P (see [10])
k-Approval, $\lvert C'\rvert \leq 2$	in P (see [9, 10])
k-Approval, $k \geq 3$, $\lvert C'\rvert \geq 3$	NP-complete (see [9, 10])
$(\alpha_i - \alpha_{i+1}) \leq (\alpha_{i+1} - \alpha_{i+2})$, $1 \leq i \leq m-2$	in P (see [9])
$(3, 2, 1, 0, \ldots, 0)$, $\lvert C'\rvert = 1$	NP-complete (see [9])

Table 1: Previous results on the complexity of PcWNA for pure scoring rules.

Theorem 3.1. *PcWNA is NP-complete for pure scoring rules of the form $(\alpha_1, \alpha_2, 1, 0, \ldots, 0)$ with $\alpha_1 > \alpha_2 > 1$, if one new candidate is added.*

Proof. Membership in NP is obvious, and the proof of NP-hardness is by a reduction from the NP-complete 3-DM problem, which is defined as follows (see [17]):

Name: Three-Dimensional Matching (3-DM).
Given: A set $M \subseteq W \times X \times Y$, with $W = \{w_1, \ldots, w_q\}$, $X = \{x_1, \ldots, x_q\}$, and $Y = \{y_1, \ldots, y_q\}$.
Question: Is there a subset $M' \subseteq M$ with $|M'| = q$, such that no two elements of M' agree in any coordinate?

Let $M \subseteq W' \times X' \times Y'$ be an instance of 3-DM with $W' = \{w'_1, \ldots, w'_q\}$, $X' = \{x'_1, \ldots, x'_q\}$, and $Y' = \{y'_1, \ldots, y'_q\}$, where $m = |M|$. Let $p(s)$ be the number of elements in M in which $s \in W' \cup X' \cup Y'$ occurs.

Construct an instance of the PcWNA problem with the election (C, V) having the set $C = W \cup X \cup Y \cup \{b, c\} \cup D$ of candidates, with $W = \{w_1, \ldots, w_q\}$, $X = \{x_1, \ldots, x_q\}$, and $Y = \{y_1, \ldots, y_q\}$. The new candidate to be added is a, so $C' = \{a\}$. D contains only dummy candidates, needed to pad the votes so as to make the reduction work. Table 2 shows the list $V = V_1 \cup V_2 \cup V_3 \cup V_4$ of votes. Note that only the first three candidates of each vote will be specified, since all other candidates do not receive any points. The numbers behind each vote denote their multiplicity. All places that need to be filled by a dummy candidate will be indicated by d (with no explicit subscript specified). Note that it is possible to substitute the d's by a polynomial number of dummy candidates such that none of them receives more than $q\alpha_1$ points.

V_1	$w_i > x_j > y_k$	1, $\forall (w'_i, x'_j, y'_k) \in M$
V_2	$w_i > d > d$	$q + m + 1 - p(w'_i)$, $\forall w_i \in W$
	$d > d > x_i$	$(q+m)\alpha_1 + (2 - p(x'_i))\alpha_2 - 1$, $\forall x_i \in X$
	$d > d > y_i$	$(q+m)\alpha_1 + \alpha_2 + 1 - p(y'_i)$, $\forall y_i \in Y$
V_3	$c > d > d$	$q + m$
	$d > c > d$	1
V_4	$d > d > b$	$(q + 2m)\alpha_1 + 2\alpha_2$

Table 2: Construction for the proof of Theorem 3.1.

The scores of the single candidates in election (C, V) are:

$$\begin{array}{rcl} score_{(C,V)}(c) & = & (q+m)\alpha_1 + \alpha_2, \\ score_{(C,V)}(w_i) & = & (q+m+1)\alpha_1, \quad 1 \le i \le q, \\ score_{(C,V)}(x_i) & = & (q+m)\alpha_1 + 2\alpha_2 - 1, \quad 1 \le i \le q, \\ score_{(C,V)}(y_i) & = & (q+m)\alpha_1 + \alpha_2 + 1, \quad 1 \le i \le q, \\ score_{(C,V)}(b) & = & (q+2m)\alpha_1 + 2\alpha_2, \\ score_{(C,V)}(d) & < & (q+m)\alpha_1 + \alpha_2, \quad \forall d \in D. \end{array}$$

Note that $score_{(C,V)}(d) < score_{(C,V)}(c)$ for all dummy candidates $d \in D$.

We claim that c is a possible winner (i.e., a can be inserted such that c wins in the election held over the candidates $C \cup C'$) if and only if there is a matching M' for the 3-DM instance M.

($\Leftarrow$) Assume that there exists a matching M' for M. Extend the votes in V to V', where a is inserted at a position with zero points in all votes of V_2 and V_3, and the votes in V_1 and V_4 are extended as shown in Table 3:

V_1	$a > w_i > x_j > y_k$	$1, \forall (w'_i, x'_j, y'_k) \in M'$
	$w_i > x_j > y_k > a$	$1, \forall (w'_i, x'_j, y'_k) \in M \setminus M'$
V_4	$d > d > a > b$	$m\alpha_1 + \alpha_2$
	$d > d > b > a$	$(q+m)\alpha_1 + \alpha_2$

Table 3: Showing ($\Leftarrow$) in the proof of Theorem 3.1.

Then all candidates except the dummy candidates have exactly $(q+m)\alpha_1 + \alpha_2$ points. Hence c has the highest score and is a winner of the election.

($\Rightarrow$) Assume that c is a winner of the election $(C \cup C', V')$, where V' is an extension of the linear votes in V. This implies that the score of all other candidates in this election is less than or equal to the score of c. The score of c will always be $(q+m)\alpha_1 + \alpha_2$, since c gets all of his or her points from the voters in V_3, where he or she is placed at the top position in $m + q$ votes and at second position in one vote.

Since $score_{(C,V)}(w_i) = (q + m + 1)\alpha_1$ points, each of the candidates w_i, $1 \le i \le q$, must lose at least $\alpha_1 - \alpha_2$ points when inserting a. Due to the requirement that $\alpha_1 > \alpha_2$, each w_i has to take at least one second position in a vote where he or she was ranked first originally. For the candidates x_i, $1 \le i \le q$, we have $score_{(C,V)}(x_i) = (q + m)\alpha_1 + 2\alpha_2 - 1$. Again, since $\alpha_2 > 1$, each x_i must lose at least $\alpha_2 - 1$ points, and since $score_{(C,V)}(y_i) = (q+m)\alpha_1 + \alpha_2 + 1$, each y_i must lose at least one point so as to not beat c.

The new candidate a can get at most $(q + m)\alpha_1 + \alpha_2$ points, since otherwise a would beat c.

To prevent w_i, $1 \le i \le q$, from beating c, a must be placed in a first position in q votes of V_1 or V_2. Then a can get at most $m\alpha_1 + \alpha_2$ points from the remaining votes without beating c. In the current situation, b would beat c by $m\alpha_1 + \alpha_2$ points. So a must take $m\alpha_1 + \alpha_2$ third positions in these votes such that b has a score of $(q+m)\alpha_1 + \alpha_2$. Then the score of a is $(q + m)\alpha_1 + \alpha_2$. Since we assumed that c is a winner of the election, every x_i, $1 \le i \le q$, must end up having $\alpha_2 - 1$ points less, and every y_i, $1 \le i \le q$, must end up having one point less. This is possible only if a is at the first position in some vote from V_1. Hence the q first positions of a must shift every candidate x_i and y_i by one position to the right. Then the triples corresponding to the three elements w_i, x_j, and y_k corresponding to these q votes must form a matching for the 3-DM instance M. ❑

3.2 Weighted Voters

In this section we study the case of weighted voters for the PcWNA problem. Obviously, all NP-hardness results obtained for PcWNA in the case of unweighted voters also hold in the case of weighted voters. However, the polynomial-time algorithms for the case of unweighted voters cannot directly be transferred to the weighted-voters case. In fact, we will show NP-hardness of PcWNA in the weighted case for some voting rules where this problem is known to be polynomial-time solvable in the unweighted case. Specifically, we will consider the plurality rule for weighted voters in this section. For plurality, polynomial-time algorithms are known for PW in the case of unweighted voters, and for Manipulation both in the unweighted-voters and in the weighted-voters case. In contrast, we now show that PcWNA is NP-complete for plurality in the case of weighted voters, even if there are only two initial candidates and one new candidate to be added.

Theorem 3.2. PcWNA *is* NP-*complete for plurality in the case of weighted voters, even if there are only two initial candidates and one new candidate to be added.*

Proof. Membership in NP is obvious. To show NP-hardness of PcWNA for plurality in the case of weighted voters, we now give a reduction from the NP-complete Partition problem, which is defined as follows (see [17]):

Name: Partition.
Given: A nonempty, finite sequence $(s_1, s_2, \ldots, s_n)$ of positive integers.
Question: Is there a subset $A' \subset A = \{1, 2, \ldots, n\}$ such that

$$\sum_{i \in A'} s_i = \sum_{i \in A \setminus A'} s_i \, ?$$

For a given Partition instance $(s_1, \ldots, s_n)$, let $\sum_{i \in A} s_i = 2K$, where $A = \{1, 2, \ldots, n\}$. We construct an election (C, V) with the set of candidates $C = \{c, d\}$, where c is the distinguished candidate, and the list of votes $V = V_1 \cup V_2$ with the corresponding weights as shown in Table 4.

V_1	$c > d$	one vote of weight K
V_2	$d > c$	one vote of weight s_i for each $i \in A$

Table 4: Construction for the proof of Theorem 3.2.

The new candidate to be added is a, so $C' = \{a\}$. In the initial situation, the score of candidate c is K, and candidate d receives $2K$ points and hence wins the election. We now show that c can be made a winner by introducing candidate a into the election if and only if there is a partition for the given Partition instance.

($\Leftarrow$) Assume that there is a subset $A' \subset A$ such that $\sum_{i \in A'} s_i = \sum_{i \in A \setminus A'} s_i$. If the new candidate a is placed at the first position in each of those votes from V_2 that correspond to the $i \in A'$, and at the last position in all remaining votes, then the score of all three candidates is exactly K, and c is a co-winner of the election.

($\Rightarrow$) Assume that c is a winner of the election, after candidate a has been introduced. It must hold that candidates a and d receive at most K points. Hence candidate d must lose K points due to inserting candidate a. This is possible only if a is placed at the first position in some votes from V_2 with a total weight of K. These votes now correspond to a valid partition. ❑

Next, we study 2-approval and give in Theorem 3.3 a result for the case of weighted voters and an unbounded number of candidates.

THEOREM 3.3. PCWNA *is* NP-*complete for* 2-*approval in the case of weighted voters, where the number of candidates is unbounded and one new candidate is to be added.*

PROOF. To prove the problem NP-hard, we again give a reduction from PARTITION, which was defined in the proof of Theorem 3.2. Let $(s_1, \ldots, s_n)$ be an instance of PARTITION with $\sum_{i \in A} s_i = 2K$, where $A = \{1, 2, \ldots, n\}$.

We introduce a set C of $n + 3$ candidates:

- c (the candidate we want to win),
- b (the candidate who wins the original election), and
- a set $\{d_0, d_1, \ldots, d_n\}$ of dummy candidates.

The votes are specified as follows:

- For each s_j, we define a vote $d_j > b > \overline{C}$ with weight s_j, where $\overline{C}$ denotes the set of candidates not yet mentioned in the vote, so in this case we have $\overline{C} = C \setminus \{b, d_j\}$. Note that the ranking of the candidates $\overline{C}$ cannot influence the outcome of the election, since we deal with 2-approval.
- There is one vote $c > d_0 > \overline{C}$ with weight K.

Since $\sum_{j \in A} s_j = 2K$, candidate b has a score of $2K$ and wins the election.

We now prove that c can be made a winner by adding one new candidate, a, if and only if there is a subset $A' \subset A$ that induces a valid partition for the given instance.

($\Leftarrow$) Suppose we have a partition $A' \subset A$. By putting a in the first position of each vote having a weight of s_i and for which $i \in A'$, a will get exactly K points. Furthermore, b loses these K points, since he or she moves to the third position in these votes. Now there is a tie between a, b, c, and d_0, each having K points. Since $s_j \leq K$, $1 \leq j \leq n$, no candidate d_j, $1 \leq j \leq n$, has a higher score. Thus, c is a co-winner of the election.

($\Rightarrow$) Suppose that c can be made a winner by adding candidate a. It follows that b has to lose at least K points. Hence, a has to be added in the votes of the form $d_j > b > \overline{C}$ at first or second position. Thus, a gets each point that b loses. But since c is made a winner by inserting a, the new candidate a can get no more than K points. Therefore, we have to insert a in a subset of votes such that the weights of these votes sum up to exactly K. Consequently, there exists a partition.

Since PARTITION is NP-complete, this proves NP-hardness. Membership in NP is straightforward. Thus PCWNA is NP-complete for 2-approval. ❑

It is easy to see that the proof of Theorem 3.3 can be transferred to k-approval: In each vote $k-2$ dummy candidates are added in the first $k-2$ positions, which gives a total number of $(k-1)(n+1)+2$ initial candidates and one new candidate. Thus we can state the following corollary.

COROLLARY 3.4. PCWNA *is* NP-*complete for k-approval in the case of weighted voters where the number of candidates is unbounded and one new candidate is to be added.*

Note that, in Corollary 3.4, the k in k-approval cannot depend on the number of candidates, since the proof is for an *unbounded* number of candidates. Table 5 summarizes the results of this section.

Scoring rule	PCWNA
Plurality, $\lvert C\rvert = 2$, $\lvert C'\rvert = 1$	NP-complete
k-Approval, $\lvert C'\rvert = 1$	NP-complete

Table 5: New results on the complexity of PcWNA in the case of weighted voters.

4. UNCERTAINTY ABOUT THE VOTING SYSTEM

4.1 Scoring Rules

In this section we study the POSSIBLE WINNER UNDER UNCERTAIN VOTING SYSTEM problem with respect to the class of scoring rules. Recall that c is the distinguished candidate we want to make a winner in the given m-candidate election, by specifying the values α_i of the scoring vector $(\alpha_1, \ldots, \alpha_m)$ appropriately. In the proof of Theorem 4.3 below we will need the following notions.

DEFINITION 4.1. *For an election $E = (C, V)$, let $pos_i(x)$ denote the total number of times candidate $x \in C$ is at position i, $1 \leq i \leq |C|$, in the list V of votes, and for all $a \in C \setminus \{c\}$, let $plus_{(c,i)}(a) = pos_i(a) - pos_i(c)$.*

If the election is held under scoring vector $(\alpha_1, \ldots, \alpha_m)$, candidate c wins if and only if for each $a \in C \setminus \{c\}$, we have $\sum_{i=1}^{|C|} plus_{(c,i)}(a) \cdot \alpha_i \leq 0$ in the co-winner case. For the unique-winner case, replace the zero on the right-hand side of the inequality by one.

In the following lemma we will show how to construct a list of votes for given values $plus_{(c,i)}(a)$ under some conditions. Let $M_{(d,i)}$ denote a circular block of $|C|-1$ votes, where candidate d is always at position i and all other candidates take all the remaining positions exactly once, by shifting them in a circular way. For example, for the set $C = \{d, c_1, \ldots, c_m\}$ of candidates the circular block $M_{(d,1)}$ looks as follows:

$$\begin{array}{ccccccccccc} d & > & c_1 & > & c_2 & > & \ldots & > & c_{m-1} & > & c_m \\ d & > & c_2 & > & c_3 & > & \ldots & > & c_m & > & c_1 \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ d & > & c_m & > & c_1 & > & \ldots & > & c_{m-2} & > & c_{m-1} \end{array}$$

LEMMA 4.2. *Let C be a set of m candidates, $c \in C$ be a distinguished candidate, $d \in C$ be a dummy candidate, and let the values $plus_{(c,i)}(a) \in \mathbb{Z}$, $1 \leq i \leq m-1$, for all candidates a in $C \setminus \{c, d\}$ be given. Let $\vec{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_m)$ be an arbitrary scoring vector with $\alpha_m = 0$. One can construct in time polynomial in m a list V of votes satisfying that:*

1. *Every candidate $a \in C \setminus \{c, d\}$ has the given values $plus_{(c,i)}(a)$, $1 \leq i \leq m-1$, in election (C, V), and*
2. *candidate d cannot beat candidate c in election (C, V).*

PROOF. Let $m = |C|$ be the number of candidates. For each positive value $plus_{(c,i)}(a)$, $1 \leq i \leq m-q$, $a \in C \setminus \{c,d\}$, we construct two types of circular blocks of votes. The first block is of type $M_{(d,i)}$, except that in the vote in which candidate a is at position m, the positions of a and d are swapped. For this block it holds that $plus_{(c,i)}(a) = 1$, and all other values $plus_{(c,j)}(b)$ and $plus_{(c,j)}(a)$, $b \in C \setminus \{c,d,a\}$, $1 \leq j \leq m-1$, remain unchanged. These blocks will be added with multiplicity $plus_{(c,i)}(a)$. To ensure that candidate d has no chance to beat candidate c, we add the votes of the circular block $M_{(d,m)}$ with multiplicity $m \cdot plus_{(c,i)}(a)$. Clearly, this block does not affect the values $plus_{(c,j)}(b)$, $1 \leq j \leq m-1$, $b \in C \setminus \{c,d\}$.

If $plus_{(c,i)}(a)$ is negative, we add the block of type $M_{(d,m)}$, where the places of a and d are swapped in the vote in which a is at position i, with multiplicity $-plus_{(c,i)}(a)$. The effect is that $plus_{(c,i)}(a)$ is decreased by 1 for each of these blocks. Again, to ensure that candidate d will not be able to beat candidate c, we add the circular block $M_{(d,m)}$ with multiplicity $-plus_{(c,i)}(a) + 1$.

By construction, the values $plus_{(c,i)}(d)$, $1 \leq i \leq n$, are never positive, so obviously d has no chance to beat or to tie with c in the election whatever scoring rule will be used. Since the votes can be stored as a list of binary integers representing their corresponding multiplicities, these votes can be constructed in time polynomial in m. ❑

To make use of Lemma 4.2, we assume succinct representation of the election (see [16]) in the following theorem. As mentioned in the above proof, this means that the votes are not stored ballot by ballot for all voters, but as a list of binary integers giving their corresponding multiplicities.

THEOREM 4.3. *Let $\mathcal{S}$ be the class of scoring rules with $m \geq 4$ candidates that are defined by a scoring vector of the form $\alpha = (\alpha_1, \ldots, \alpha_{m-4}, x_1, x_2, x_3, 0)$, with $x_i = 1$ for at least one $i \in \{1,2,3\}$. $\mathcal{S}$-PcWUVS and $\mathcal{S}$-PWUVS are NP-complete (assuming succinct representation).*

PROOF. Membership in NP is obvious, and the proof of NP-hardness will be by a reduction from the NP-complete problem INTEGER KNAPSACK (see, e.g., [17]):

Name: INTEGER KNAPSACK
Instance: A finite set of elements $U = \{u_1, \ldots, u_n\}$, two mappings $s, v : U \to \mathbb{Z}^+$, and two positive integers, b and k.
Question: Is there a mapping $c : U \to \mathbb{Z}^+$ such that

$$\sum_{i=1}^{n} c(u_i)s(u_i) \leq b \text{ and } \sum_{i=1}^{n} c(u_i)v(u_i) \geq k\,?$$

We first focus on the co-winner case and then show how to transfer the proof to the unique-winner case. Let (U, s, v, b, k) be an instance of INTEGER KNAPSACK with $U = \{u_1, \ldots u_n\}$ and let $c : U \to \mathbb{Z}^+$ be a mapping. Then it holds that

$$\begin{array}{c}\sum_{i=1}^{n} c(u_i) \cdot s(u_i) \leq b \\ \sum_{i=1}^{n} c(u_i) \cdot v(u_i) \geq k\end{array} \tag{2}$$

$$\Leftrightarrow \begin{pmatrix} s(u_1) & s(u_2) & \ldots & s(u_n) \\ -v(u_1) & -v(u_2) & \ldots & -v(u_n) \end{pmatrix} \begin{pmatrix} c(u_1) \\ c(u_2) \\ \vdots \\ c(u_n) \end{pmatrix} \leq \begin{pmatrix} b \\ -k \end{pmatrix}$$

$$\Leftrightarrow \begin{pmatrix} & -b' \\ & k' \\ & nb \\ A & (n-1)b \\ & \vdots \\ & b \end{pmatrix} \begin{pmatrix} c'(u_1) \\ c'(u_2) \\ \vdots \\ c'(u_n) \\ 1 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{3}$$

$$\text{with } A = \begin{pmatrix} s(u_1) & s(u_2) & \ldots & s(u_n) \\ -v(u_1) & -v(u_2) & \ldots & -v(u_n) \\ -1 & 0 & \ldots & 0 \\ 0 & -1 & \ldots & 0 \\ \vdots & & & \\ 0 & \ldots & 0 & -1 \end{pmatrix}, \text{ where}$$

$$\begin{aligned} c'(u_i) &= c(u_i) + (n-i+1)b, \quad 1 \leq i \leq n, \\ b' &= b + \sum_{i=1}^{n} b \cdot s(u_i) \cdot (n-i+1), \text{ and} \\ k' &= k + \sum_{i=1}^{n} k \cdot v(u_i) \cdot (n-i+1). \end{aligned}$$

The last n rows of the matrix ensure that

$$c'(u_i) \geq (n-i+1)b, \quad 1 \leq i \leq n,$$

and so there are no new solutions added for which the values $c(u_i)$ may be negative. Furthermore, since $c(u_i) \leq b$, it is now ensured that $c'(u_1) \geq c'(u_2) \geq \cdots \geq c(u_n) \geq b$. Hence it still holds that c is a solution for the given INTEGER KNAPSACK instance if and only if c' is a solution for (3).

We will now build an election $E = (C, V)$ with candidate set $C = \{c, d, e, f, g_1, \ldots, g_n\}$, where c is the distinguished candidate and d is a dummy candidate who cannot beat c in the election whatever scoring rule will be used. The list of votes will be built using Lemma 4.2 according to the matrix in (3). The $n+2$ rows in the matrix correspond to the candidates e, f, and $g_1, \ldots, g_n$. Since the matrix has only $n+1$ columns, the positions $n+2$ and $n+3$ in the votes will have no effect on the outcome of the election, and thus the corresponding $plus_{(c,i)}(a)$ values, $n+2 \leq i \leq n+3$, can be set to zero for all candidates $a \in \{e, f, g_1, \ldots, g_n\}$. The corresponding values in the scoring vector can be set to either zero or one, respecting the conditions for a valid scoring vector. Hence, the votes in V have to fulfill the following properties:

$$plus_{(c,i)}(e) = \begin{cases} s(u_i) & \text{for } 1 \leq i \leq n \\ -b' & \text{for } i = n+1 \\ 0 & \text{for } n+2 \leq i \leq n+3, \end{cases}$$

$$plus_{(c,i)}(f) = \begin{cases} -v(u_i) & \text{for } 1 \leq i \leq n \\ k' & \text{for } i = n+1 \\ 0 & \text{for } n+2 \leq i \leq n = n+3, \end{cases}$$

$$plus_{(c,i)}(g_j) = \begin{cases} -1 & \text{for } 1 \leq i \leq n, i = j \\ (n-i+1)b & \text{for } i = n+1, 1 \leq j \leq n \\ 0 & \text{for } 1 \leq i \leq n+3, \\ & \quad 1 \leq j \leq n,\ i \neq j. \end{cases}$$

According to Lemma 4.2, these votes can be constructed in polynomial time such that the dummy candidate d has no influence on c being a winner of the election, whatever scoring rule of type $\alpha = (\alpha_1, \dots, \alpha_n, 1, \alpha_{n+2}, \alpha_{n+3}, 0)$ will be used.

Since the $plus_{(c,i)}(a)$ values assigned to the candidates $a \in C \setminus \{c, d\}$ are set according to the matrix in (3), it holds that c can be a winner in election $E = (C, V)$ by choosing a scoring rule of the form $\alpha = (\alpha_1, \dots, \alpha_n, 1, \alpha_{n+2}, \alpha_{n+3}, 0)$ if and only if for each $a \in C \setminus \{c\}$, we have

$$\sum_{i=1}^{n} plus_{(c,i)}(a) \cdot c(u_i) + plus_{(c,n+1)}(a) \leq 0.$$

As described above, the values in the scoring vector for positions $n+2$ and $n+3$, have no effect on the outcome of the election. Hence, by switching rows in the matrix we can extend the set of possible scoring rules to scoring rules of the form $\alpha = (c(u_1), \dots, c(u_n), x_1, x_2, x_3, 0)$, with $x_i = 1$ for at least one $i \in \{1, 2, 3\}$. Hence, c can be made a winner of the election $E = (C, V)$ if and only if there is a solution to (3). Since we have shown above that there is a solution to (2) if and only if there is a solution to (3), it holds that there is a solution c to our Integer Knapsack instance if and only if there is a scoring rule α, of the form described above, under which c wins the election $E = (C, V)$.

To see that this reduction also settles the unique-winner case, note that (3) is equivalent to the following inequality:

$$\begin{pmatrix} & -b'+1 \\ & k'+1 \\ & nb+1 \\ A & (n-1)b+1 \\ & \vdots \\ & b+1 \end{pmatrix} \begin{pmatrix} c'(u_1) \\ c'(u_2) \\ \vdots \\ c'(u_n) \\ 1 \end{pmatrix} \leq \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}. \quad (4)$$

The election we need to construct has the same candidate set as above and the voters are constructed according to the values $plus_{(c,n+1)}(a)$ for $a \in C \setminus \{c, d\}$ in the matrix of (4). Thus, c is the unique winner of the modified election if and only if for each $a \in C \setminus \{c\}$, we have

$$\sum_{i=1}^{n} plus_{(c,i)}(a) \cdot c(u_i) + plus_{(c,n+1)}(a) \leq 1.$$

By a similar argument as above, there is a scoring rule of the form $\alpha = (\alpha_1, \dots, \alpha_n, x_1, x_2, x_3, 0)$ with $x_i = 1$ for at least one $i \in \{1, 2, 3\}$ in which c wins the election if and only if there is a solution c for the given Integer Knapsack instance. ❑

4.2 Copeland$^\alpha$ Elections

In Copeland$^\alpha$ elections [14], the parameter α is a rational number from the interval $[0, 1]$ that specifies how ties are rewarded in the pairwise comparisons between candidates.

Theorem 4.4. *$\mathcal{C}$-PcWUVS and $\mathcal{C}$-PWUVS are polynomial-time solvable for the family of Copeland$^\alpha$ elections:*

$$\mathcal{C} = \{Copeland^\alpha \mid \alpha \text{ is a rational number in } [0, 1]\}.$$

Proof. To decide whether a distinguished candidate c can be made a winner of the election by choosing the parameter α after all the votes have been cast, we do the following. In the co-winner case, for each $c_i \in C \setminus \{c\}$, compute

$$f(c_i) = \begin{cases} \frac{win(c) - win(c_i)}{tie(c) - tie(c_i)} & \text{if } tie(c) \neq tie(c_i) \\ win(c) - win(c_i) & \text{otherwise.} \end{cases}$$

If $f(c_i) \geq 0$ for all $c_i \in C$, c can be made a winner of the election by setting $\alpha = \min_{c_i \in C}\{f(c_i), 1\}$, and otherwise c cannot be made a winner. So $\mathcal{C}$-PcWUVS is in P.

In the unique-winner case, for c to be the unique winner winner of the election, it must hold that $f(c_i) > 0$ and α is set to a value greater than $\min_{c_i \in C}\{f(c_i)\}$ if this value is less than one, or else to one. Otherwise, c cannot be made the unique winner of the election. So $\mathcal{C}$-PWUVS is in P. ❑

4.3 Preference-Based Approval Voting

In approval voting the situation is a bit different, since approval voting is not a class of voting systems, and the voters usually do not report linear preferences but approval vectors. Brams and Sanver [7, 8] proposed various voting systems that combine preference-based voting and approval voting. Here the voters report a strict preference order, along with an approval line indicating that the voter approves of all candidates to the left of this line and disapproves of all candidates to the right of this line. They require votes to be *admissible* [7], which means that each voter approves of his or her first ranked candidate and disapproves of his or her last ranked candidate. If we assume that the approval lines are not set by the voters (who thus only report their linear orders) but are set by the voting system itself (after all votes have been cast), we obtain (for m candidates and n voters) a class $\mathcal{A}_{m,n}$ of $(m-1)^n$ voting systems. For each such system, the candidates with the highest number of approvals win. Note that these voting systems are not very natural (as they do not let the voters themselves choose their approval strategies) and do not possess generally desirable social-choice properties (e.g., the systems in $\mathcal{A}_{m,n}$ are not even anonymous, as changing the order of votes may result in a different outcome).

In this setting, given an election where voters report their preference orders, setting the approval lines afterwards corresponds to choosing a system from $\mathcal{A}_{m,n}$. It is easy to see that PcWUVS and PWUVS are polynomial-time solvable for this class. To make the distinguished candidate c win the election, choose the system that sets the approval line in each vote that does not rank c at the last position right behind c, and in the votes that do rank c last right behind the top candidate. If c is not a winner (unique winner) of this election, c cannot win (be a unique winner of) the election whatever system from the class is chosen. Thus, PcWUVS and PWUVS are polynomial-time solvable for this class of preference-based approval voting systems.

In contrast to this result, Elkind et al. [12] show NP-hardness for a related bribery problem, even if the briber is only allowed to move the approval line.

5. CONCLUSIONS AND FUTURE WORK

For the Possible Winner problem, a full dichotomy result for the class of pure scoring rules is known [6, 5]. In contrast, the complexity of the related problem PcWNA has not yet been completely settled and the question raised by Chevaleyre et al. [9] remains open. Our result stated in Theorem 3.1 makes a further step towards this goal by showing NP-completeness of PcWNA for a whole class of pure scoring rules. An interesting task for future work would be to characterize this problem for all pure scoring rules in terms of a dichotomy result. Moreover, our initial work on weighted voters for PcWNA might be extended, and for both the weighted and the unweighted case the unique-winner variant PWNA should be further explored (see also [9, 25]). Another problem also stated in [9] concerns the number of new candidates to be added. Up to now NP-hardness results for pure scoring rules are known only for the case where one new candidate is added. What about adding more than one candidate? Note that the problem becomes easy if an unbounded number of new candidates is to be added.

For the PcWUVS and PWUVS problems, the next obvious step would be to extend Theorem 4.3 to unrestricted scoring rules, ideally with the goal of obtaining a complete dichotomy result. It would also be interesting to study these problems for other natural classes of voting systems, for example, for all voting systems sharing some important social-choice property (e.g., for all Condorcet systems).

Acknowledgment

We thank the reviewers for the very helpful comments. This work was supported in part by DFG grants RO 1202/{11-1, 12-1} and the ESF EUROCORES program LogICCC.

6. REFERENCES

[1] J. Bartholdi III and J. Orlin. Single transferable vote resists strategic voting. *Social Choice and Welfare*, 8(4):341–354, 1991.

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

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

[4] D. Baumeister, G. Erdélyi, E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Computational aspects of approval voting. In J. Laslier and R. Sanver, editors, *Handbook on Approval Voting*, chapter 10, pages 199–251. Springer, 2010.

[5] D. Baumeister and J. Rothe. Taking the final step to a full dichotomy of the possible winner problem in pure scoring rules. In *Proceedings of the 19th European Conference on Artificial Intelligence*, pages 1019–1020. IOS Press, Aug. 2010. Short paper.

[6] N. Betzler and B. Dorn. Towards a dichotomy of finding possible winners in elections based on scoring rules. In *Proceedings of the 34th International Symposium on Mathematical Foundations of Computer Science*, pages 124–136. Springer-Verlag *Lecture Notes in Computer Science #5734*, Aug. 2009.

[7] S. Brams and R. Sanver. Critical strategies under approval voting: Who gets ruled in and ruled out. *Electoral Studies*, 25(2):287–305, 2006.

[8] S. Brams and R. Sanver. Voting systems that combine approval and preference. In S. Brams, W. Gehrlein, and F. Roberts, editors, *The Mathematics of Preference, Choice, and Order: Essays in Honor of Peter C. Fishburn*, pages 215–237. Springer, 2009.

[9] Y. Chevaleyre, J. Lang, N. Maudet, and J. Monnot. Possible winners when new candidates are added: The case of scoring rules. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, pages 762–767. AAAI Press, July 2010.

[10] Y. Chevaleyre, J. Lang, N. Maudet, J. Monnot, and L. Xia. New candidates welcome! Possible winners with respect to the addition of new candidates. Technical Report Cahiers du LAMSADE 302, Université Paris-Dauphine, 2010.

[11] V. Conitzer, T. Sandholm, and J. Lang. When are elections with few candidates hard to manipulate? *Journal of the ACM*, 54(3):Article 14, 2007.

[12] E. Elkind, P. Faliszewski, and A. Slinko. Swap bribery. Technical Report cs.GT/0905.3885, ACM Computing Research Repository (CoRR), May 2009.

[13] E. Elkind, P. Faliszewski, and A. Slinko. Cloning in elections. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, pages 768–773. AAAI Press, July 2010.

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

[15] P. Faliszewski, E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. A richer understanding of the complexity of election systems. In S. Ravi and S. Shukla, editors, *Fundamental Problems in Computing: Essays in Honor of Professor Daniel J. Rosenkrantz*, pages 375–406. Springer, 2009.

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

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

[18] A. Gibbard. Manipulation of voting schemes. *Econometrica*, 41(4):587–601, 1973.

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

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

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

[22] M. Pini, F. Rossi, K. Venable, and T. Walsh. Dealing with incomplete agents' preferences and an uncertain agenda in group decision making via sequential majority voting. In *Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning*, pages 571–578. AAAI Press, Sept. 2008.

[23] M. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2):187–217, 1975.

[24] T. Walsh. Uncertainty in preference elicitation and aggregation. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, pages 3–8. AAAI Press, July 2007.

[25] L. Xia, J. Lang, and J. Monnot. Possible winners when new alternatives join: New results coming up! In V. Conitzer and J. Rothe, editors, *Proceedings of the 3rd International Workshop on Computational Social Choice*, pages 199–210. Universität Düsseldorf, Sept. 2010.