The Complexity of Losing Voters

Tomasz Perek and Piotr Faliszewski
AGH University of Science and Technology
Krakow, Poland
mat.dexiu@gmail.com,
faliszew@agh.edu.pl

Maria Silvia Pini and Francesca Rossi
University of Padova
Padova, Italy
pini@dei.unipd.it, frossi@math.unipd.it

ABSTRACT

We consider the scenario of a parliament that is going to vote on a specific important issue. The voters are grouped in parties, and all voters of a party vote in the same way. The expected winner decision is known, because parties declare their intentions to vote, but before the actual vote takes place some voters may leave the leading party to join other parties. We investigate the computational complexity of the problem of determining how many voters need to leave the leading party before the winner changes. We consider both positional scoring rules (plurality, veto, k-approval, k-veto, Borda) and Condorcet-consistent methods (maximin, Copeland), and we study two versions of the problem: a pessimistic one, where we want to determine the maximal number of voters that can leave the leading party without changing the winner; and an optimistic one, where we want the minimal number of voters that must leave the leading party to be sure the winner will change. These two numbers provide a measure of the threat to the expected winner, and thus to the leading party, given by losing some voters. We show that for many positional scoring rules these problems are easy (except for the optimistic version with k-approval, for k at least 3, and Borda). Instead, for Condorcet-consistent rules, they are both computationally difficult, with both Maximin and Copeland.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*; F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity

General Terms

Theory, Algorithms

Keywords

Voting protocols, computational complexity, manipulation, control.

1. INTRODUCTION

We consider a parliament, where voters are grouped into parties and where some particularly important vote is to be taken (e.g., deciding on the country's budget for the coming year, or deciding whether the current prime minister should stay in the office or if a new one—and who should that be—should be chosen). Since the election at hand is such an important one, we assume that the parties require all their members to follow party discipline, that is, each party requires its members to vote in a certain given way. We assume that there is one particularly strong party (we call it the leading party) that has sufficiently many members that its top preferred candidate is winning. (In practice, this would be the party forming the government.) However, when such important elections happen in a parliament, some not-so-happy members of the leading party (perhaps those that have been mistreated in the past or who do not fully identify with the party's politics) try to negotiate with their party leader to obtain some benefits for themselves, threatening that if not satisfied, they will join other parties. In this paper we take the position of the leading party's leader, who has to decide how many/which of these party members he should satisfy.

More precisely, we assume that before an election takes place it is known how all parties will vote, so the winner is known, but the loss of some of the voters by the leading party may result in a different winner. The question we are asking is: How safe is the leading party with respect to this kind of actions? In particular, it would be interesting for such a leading party to know the maximal number of voters that it may lose without posing any threat to the winner of the election (or, in other words, the minimal number of voters that must leave the leading party in order to have a chance of changing the winner). This will be called the *pessimistic* variant of the question we are addressing. The other interesting bound for a leading party is the minimal number of voters whose loss would certainly change the winner, regardless of which parties they decide to join (or, in other words, the maximal number of voters that may leave the leading party so that there would still be a chance of keeping the winner unaltered). This will be called the *optimistic* variant of our question. These two numbers (that is, the answers to the pessimistic and the optimistic variants) provide a measure of the stability of the current winner of the election.

Figure 1 shows a graphical representation of the stability of the winner. *Siz* is the overall number of voters of the leading party, while *Pes* and *Opt* are the numbers corresponding to the pessimistic and optimistic variant solutions, respectively. If the leading party loses less than *Pes* voters, the winner is safe; if it loses between *Pes* and *Opt* voters, the winner is threatened, that is, it could change; if it loses more than *Opt* voters, the winner will surely change.

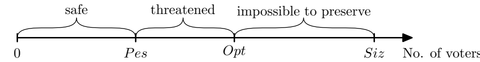


Figure 1: Status of the winner depending on the number of voters leaving the leading party.

For example, let us consider a parliament with 5 parties, P_1 through P_5, where P_1 has 8 voters, P_2 has 3 voters, and all other

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.*

parties have 2 voters. We are going to use the Plurality rule to aggregate the votes of such voters. This means that we will just consider their first choice and declare as winners those candidates that are the first choice for the highest number of voters. Assume we have 4 candidates: a, b, c, and d. Assume also that P_1 declares the intention to vote for a, P_2 for c, P_3 for d, and both P_4 and P_5 for b. In this situation, P_1 is the leading party and a is the winner. Specifically, a has 8 votes, b has 4 votes, c has 3 votes, and d has 2 votes. If we move up to 2 voters from P_1 to any combination of the other parties, a will still be a winner: If we move 2 voters, a will still have 6 points, and even if both voters go to P_4 or P_5 (thus giving b 6 votes in total), a would still be among the winners. Thus the winner is safe up to the loss of 2 voters. If instead we move 3 voters, then the winner could change, so we are in the threatened range of Figure 1. In fact, if these 3 voters, who are currently in P_1, all move to P_2, then a will lose, since it will get 5 votes and c will get 6 votes. Instead, if 2 voters move to P_2 and 1 voters moves to either P_4 or P_5, then a, b, and c will all get 5 votes and d will get 2 votes, so a will still be among winners. With the loss of 4 or more voters from P_1, the winner will always change. Indeed, if 4 voters leave P_1, a will get just 4 votes, and the 4 votes will be given to b, c, and d, that now have, respectively, 4, 3, and 2 votes. There is no way that all of them will have at most 4 votes (there is space only for $1+2$ votes). So, in this example, $Pes = 2$ and $Opt = 4$.

In this paper we study the computational complexity of finding the answer to both variants of the considered problem, for various election systems. We show that the pessimistic problem is easy for all scoring protocols, but that the complexity of the optimistic one depends on the scoring rule (for example, it is easy for Plurality, 2-approval, and k-veto, but is NP-complete for k-approval where $k \geq 3$ and for Borda). On the other hand, for the two Condorcet-consistent rules that we study (i.e., for Copeland and Maximin), both problems are computationally hard. Naturally one could study other voting rules as well. We have chosen these ones as they are natural, representative members of scoring rules and Condorcet consistent rules (though, for example, our analysis misses elimination-based rules such as STV or Nanson's rule).

Although the terms used above are related to political elections, the same scenarios can occur also in other contexts. Elections may be used by solvers to find the best solution for a problem, by a group of friends to determine what to have for dinner, by a group of radio listeners to decide for their favourite song, as well as by search engines to determine the ranking of the most popular websites and to avoid spam [4,7], or also by routers to designate one of them in a routing protocol. Elections have been used also in various tasks of collaborative filtering [16] (that is, techniques of filtering large sets of data for patterns and information of special interest of users, usually involving collaboration among multiple agents and data sources), as well as in planning tasks for automated multiagent systems [5,6]. At least in some of these contexts our problem still makes sense. For example, if several solvers are trying to find the best solution to a problem (and these solvers have weights measuring the amount of trust we have in them and they vote on the final solution) then the amount of weight that we need to move away from the solver that currently dictates the solution so that the result changes (or, may change) can be viewed as measuring the amount of confidence we should have in the solution.

The paper is organized as follows. In Section 2 we provide the background notions, Sections 4 and 5 contain the results on scoring rules and Condorcet-consistent rules. We present related work in Section 6. Finally, in Section 7 we summarize our results and propose lines for future work.

2. BACKGROUND NOTIONS

In this section we introduce the main notions that we use throughout the paper. More detailed definitions and results about voting theory can be found in [2]. An *election* is a pair $E = (C,V)$, where $C = \{c_1,\dots,c_m\}$ is a set of *candidates* and $V = \{v_1,\dots,v_n\}$ is a set of *voters*. Each voter v_i is represented via its preference order $>_i$, which is a strict linear order over the candidates in C. Occasionally, when specifying a preference order, we will include a set in the preference order. This means ranking the candidates in this set in some arbitrary but fixed order. To indicate the reverse of this fixed order, we put an arrow over the set.

An *election system* (or voting rule) R maps an election $E = (C,V)$ to a nonempty set W, $W \subseteq C$, of *winners*. We assume the nonunique-winner model, i.e., all members of $R(E)$ are considered to be winning.

A *scoring protocol* for m candidates is an election system defined by a vector $\alpha = (\alpha_1,\alpha_2,\dots,\alpha_m)$ (so called scoring vector) of non-negative integers such that $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_m$. A candidate on the i-th place on a voter's preference list receives α_i points from that voter. Examples of scoring protocols (for m candidates), that will be considered in this paper, are: Plurality (scoring vector $(1,0,\dots,0)$), Veto (scoring vector $(1,1,\dots,1,0)$), k-approval (scoring vector with k ones followed by $m-k$ zeros), and Borda's rule (scoring vector $(m-1,m-2,\dots,0)$). We write k-veto to mean $(m-k)$-approval.

A *Condorcet-consistent rule* is an election system that always elects the Condorcet winner, if it exists. The Condorcet winner is a candidate who wins by majority in all pairwise elections with any other candidate. Examples of Condorcet consistent rules, that will be considered in this paper, are Maximin and Copeland. The Copeland rule associates to each candidate a score which is the number of other candidates it wins in pairwise competitions, and the winners are the candidates with the largest score. In this version of the Copeland rule, a win in a pairwise competion counts 1 for the winner and 0 for the loser, while a tie counts 0 for both. Other versions of the Copeland rule treat ties differently, counting them as α points for both candidates, where α usually is between 0 and 1. When we will need to indicate a specific α, we will write Copeland$^{\alpha}$ to mean the Copeland rule that uses such α. The Maximin rule records, for each pairwise competition, not only the winner, but also the majority (that is, the number of voters who make it a winning competition). Each candidate is then associated to a score which is the minimum majority (or minority) on all its pairwise competitions with all other candidates. The winners are then the candidates with the largest score. More formally, let $E = (C,V)$ be an election. For two distinct candidates c and d, we define $N_E(c,d) = \|\{v \in V \mid v \text{ prefers } c \text{ to } d\}\|$ and $M_E(c,d) = N_E(c,d) - N_E(d,c)$. We define Maximin score of candidate c to be $\min_{d \in C-\{c\}} M_E(c,d)$. (We point out that many authors take the minimum over N_E and not M_E, but our definition is equivalent because we assume strict total orders as voters' preferences. Our definition is easier to work with in our proofs.)

For the voting rules that assign scores, we write $\text{score}_E(c)$ to denote the score of candidate c in election E. The particular election system will always be clear from context.

Our hardness proofs follow by reductions from the X3C problem [11]. We recall here the formal definition of this NP-complete problem.

DEFINITION 1. *In the X3C problem we are given a set* $B = \{b_1,b_2,\dots,b_{3t}\}$, $t > 1$, *and a family* $\mathscr{S} = \{S_1,S_2,\dots,S_m\}$ *of three-element subsets of B. We ask if there is a set* $A \subseteq \{1,2,\dots,m\}$ *such that* $|A| = t$ *and* $\bigcup_{i\in A} S_i = B$.

3. PROBLEM STATEMENT

The election scenario we consider is one where voters are grouped into *parties*. That is, in addition to the set of voters $V = (v_1, \ldots, v_n)$ we have a set $\mathscr{P} = \{P_1, \ldots, P_k\}$ of parties, where $P_1, \ldots, P_k$ are mutually disjoint subsets of V. We assume that all voters in the same party vote in the same way. An *election with parties* is therefore a triple $E = (C, V, \mathscr{P})$.

We assume that a *leading party* is given as part of the input and that the candidate that is most preferred by this party wins the election (or is one of the winners). Without loss of generality, we also assume that the given leading party is P_1.

Finally, we define a *losing function* $l : V \to \mathscr{P}$ that for each voter indicates which party he moves to. Losing functions have the following property: Each voter who belongs to parties $\{P_2, \ldots, P_k\}$ stays in his or her respective party, but the voters from P_1 may move to the other parties. We will study the computational complexity of determining:

1. The maximal number of voters that can leave from the leading party without changing the set of winners, regardless of which parties they choose to join. We will call this the *pessimistic* variant of the voter losing problem.
2. The minimal number of voters that must leave the leading party to ensure that the winning candidate will not win, regardless of which parties they choose to join. We will call this the *optimistic* variant of the voter losing problem.

More formally, we are given: an election with parties $E = (C, V, \mathscr{P})$ with a set of n voters $V = (v_1, \ldots, v_n)$ in k parties $\mathscr{P} = \{P_1, \ldots, P_k\}$, where P_1 is the leading party, a set of m candidates $C = \{c_1, \ldots, c_m\}$, an election system R, and a winner w of the election E (i.e., $w \in R(E)$; w is the top choice of P_1). We then consider the following two decision problems:

R-pessimistic-voter-losing: Given a natural number q, is there a set $T \subseteq P_1$ with $|T| \geq q$ such that, for all losing function $l : V \to \mathscr{P}$, after moving every voter $v \in T$ to the party indicated by $l(v)$, w would still be a winner?

R-optimistic-voter-losing: Given a natural number q, is there a set $T \subseteq P_1$ with $|T| \leq q$ such that, for all losing functions $l : V \to \mathscr{P}$, after moving every voter $v \in T$ to the party indicated by $l(v)$, w would not be a winner?

It is easy to show that the following result holds.

THEOREM 2. *If the number of parties is a fixed constant and R is an anonymous voting rule with polynomial-time winner determination procedure, then both R-pessimistic-voter-losing and R-optimistic-voter-losing are in* P.

Since in parliaments one typically expects to have few candidates, one could say that this theorem is "the end of the story" for our problem. However, the proof of this theorem relies on the fact that with n voters in the leading party and k parties in total, we can consider all n^k distributions of P_1 members among the parties. Under our assumption this mean polynomial running time, but in practice such complexity may be prohibitive. In consequence, polynomial-time algorithms that we provide later are much better for their settings (or, in case of optimistic setting for k-veto are, at least, somewhat better) and computational hardness results justify when we are bound to use this brute-force algorithm.

4. SCORING PROTOCOLS

In this section we present our complexity results regarding the voter losing problems with respect to scoring protocols.

4.1 Pessimistic Voter Losing

We show that for every scoring protocol the voter losing problem in its pessimistic variant is computationally easy.

THEOREM 3. *For every given scoring protocol, pessimistic-voter-losing is in* P. *This holds even if the scoring protocol is part of the input.*

PROOF. We give an algorithm for our problem. Let $E = (C, V, \mathscr{P})$ be our input election, where $P_1 \in \mathscr{P}$ is the leading party and let w be the leading party's top candidate (w is a winner in E). For each party $P \in \mathscr{P}$ and each candidate $c \in C$, by $\mathrm{score}_E^P(c)$ we mean the number of points that candidate c gets from a single voter from party P.

If all parties place w on top of their preference lists, then clearly it is possible to move all the voters away from P_1 without preventing w from winning. Otherwise, let A be the set of all those candidates who, for some party, are placed on a position where they get more points than w. For each candidate $c \in A$ and for each party P_j (apart from the leading party P_1) we compute the expression

$$(\mathrm{score}_E^{P_j}(c) - \mathrm{score}_E^{P_j}(w)) - (\mathrm{score}_E^{P_1}(c) - \mathrm{score}_E^{P_1}(w)),$$

choosing the party, for which it has the greatest possible value (we use $\rho(c)$ to denote this value and $\psi(c)$ to denote corresponding party P_j). Note that since we have limited ourselves to only those candidates, who are not "dominated" by w on at least one preference list—the greatest value for the above expression is guaranteed to be positive. This is because the first term of the subtraction for that party (ranking the currently considered candidate over w) is positive and the second term is nonpositive for all parties and candidates. Now, for each candidate $c \in A$ we compute $\phi(c) = \lceil \frac{\mathrm{score}_E(w) - \mathrm{score}_E(c)}{\rho(c)} \rceil$ and choose that candidate (let us call him x) for whom the value of this expression is the smallest. It remains to note that $\phi(x)$ is the smallest number of voters whose move away from P_1 can change the winner (provided they move to $\psi(x)$). Clearly, the is polynomial-time and is correct. □

COROLLARY 4. *R-Pessimistic-voter-losing is in* P *when R is Plurality, k-veto, k-approval, and Borda.*

4.2 Optimistic Voter Losing

Let us now move on to the optimistic setting. Here we seek the maximum number of voters that we can move away from the leading party before its top candidate ceases to be a winner.

Plurality. Not surprisingly, for Plurality the problem is easy.

THEOREM 5. *Plurality-optimistic-voter-losing is in* P.

PROOF. To prove the theorem, we provide a polynomial-time algorithm that will not change the result of the election for as many moved voters as possible. Let w be the top candidate of the leading party. If there exists another party P_i (apart from the leading one) that supports w, then we can move every voter to that party and w would still be a winner. Otherwise, let A be the set of candidates who are top choice of at least one party (except for the leading party). It suffices to move voters to a party which votes for the least popular candidate from set A *at the moment* (i.e., the candidate from A with the fewest points at the moment of choosing him). In this way we will keep w winning for as long as possible. The algorithm works in polynomial time since we will move at most $|P_1|$ voters and for each of them we can determine the target party of the transfer in polynomial time. □

k-approval and k-veto. Intuitively, k-approval (k-veto) for $k > 2$ does not seem to be a natural voting rule for a parliament because it is not unanimous. Even if all the members of parliament agreed on a preference order, they would still select k top choices ($m-k$ top choices, where m is the number of alternatives) rather than the single top one. Nonetheless, we provide the analysis of this family of rules for a number of reasons: (a) our study is not limited to parliaments, (b) to allow comparison with other voting problems, and (c) because technical ideas developed in this section might be interesting in other settings.

We start with k-approval. We prove that the optimistic variant is easy for $k=2$, while it is intractable for any value of k that is greater than 2. We assume that $k \geq 2$, since for $k=1$ the problem is identical to the one presented for Plurality. We prove tractability of the optimistic variant of voter losing problem for 2-approval by exploiting its similarity to a certain polynomial-time solvable matching problem.

THEOREM 6. *2-approval-optimistic-voter-losing is in* P.

PROOF. Let w be the top candidate in the leading party's (i.e., P_1's) preference order (prior to any voter movement, w is among the winners). Without loss of generality, we can assume that there is no other party that supports the same two candidates as the leading party (otherwise we can move every voter to that party and w would still be a winner). Let us assume that exactly $k = k' + k''$ voters move from the leading party P_1 to the other ones, where k' is the number of voters who move to the parties that do *not* support w, and k'' is the number of voters who move to the parties that *do* support w. Candidate w will have $\text{score}_{E'}(w) = \text{score}_E(w) - k'$ points. Let $s : C \to \mathbb{Z}$ be a function defined as follows: $s(w) = k''$ and for every candidate $x \in C \setminus \{w\}$, $s(x) = \text{score}_{E'}(w) - \text{score}_E(x) + b(x)$, where

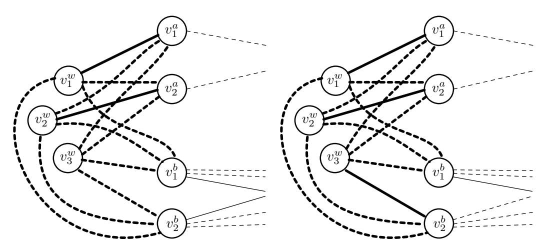
$$b(x) = \begin{cases} k' + k'', \text{ if } P_1 \text{ supports } x \\ 0, \text{ otherwise} \end{cases}$$

E' will denote the transformed election that we get from E after moving all voters according to the description provided in this proof. Function s indicates how many more voters at the beginning may vote for the particular candidate before he or she defeats w (and, in case of w, function s indicates how many more voters *must* vote for him).

Now we construct a graph whose vertices and edges will in some way correspond to candidates and parties, respectively. For each candidate $x \in C$ (including w) we add to the graph exactly $s(x)$ corresponding vertices. For each party P_i (apart from P_1) we add edges as follows: Supposing that P_i supports candidates q and r, we add to the graph edges that connect each of the vertices corresponding to candidate q with each of the vertices corresponding to candidate r (thus creating a local *complete bipartite graph*). Since we assumed that there is no other party supporting the same candidates as the leading one and we explicitly excluded P_1 from the routine above, there is no edge connecting vertices corresponding to candidate w with the other candidate supported by P_1. Finally, we assign weights to the edges of the graph with a function $\omega : E \to \mathbb{Z}$ defined as follows: $\omega(e) = 2$ if $w \in e$, and $\omega(e) = 1$ otherwise. For such a graph, we solve (in polynomial time) a maximum weighted matching problem (see, e.g., [14]), whose solution corresponds almost directly to the solution of the 2-approval-optimistic-voter-losing problem: adding an edge to the matching indicates a single voter moving to the party corresponding to that edge. All we need to do is to check whether *exactly* k'' edges from the maximum weighted matching are incident to vertices corresponding to w and whether there are *at least* k' other edges in that matching.

CLAIM 7. *For a graph defined above, a maximum weighted matching contains as many edges incident to w-originating vertices as possible (i.e., it is not possible to 'exchange' one of the other edges of the matching for a w-incident edge).*

PROOF. Assume to the contrary that we have the graph defined above, and that—after finding a maximum weighted matching—there is an unmatched vertex corresponding to w, for example as the vertex v_3^w on Figure 2(a). Thick and thin lines denote edges of weight 2 and 1, respectively. Continuous lines denote edges in the matching, and dotted lines—the other ones. However, in that case, it would suffice to exchange some edge of weight 1 to a w-incident edge of weight 2 in order to increase the overall weight of the matching (which contradicts our assumption that the matching found is a *maximum* weighted matching). Such an exchange may take place either in a single step (as in the example on Figure 2), or as a result of a series of exchanges of adjacent edges along some path (if the 1-weighted edge being exchanged is not adjacent to any w-incident edge)—the process is known as a *matching alternation* along an alternating path [14]. Figure 2 shows an exemplary exchange: initially, the w-originating vertex v_3^w is not incident to the matching. Then, an edge incident to the vertex v_2^b (which belongs to the matching) is exchanged for the edge $\{v_3^w, v_2^b\}$. It is worth to note that the weights of the edges do not need to be exactly as specified—all we need is to ensure that w-incident edges have greater weights than the other ones. □



(a) (b)

Figure 2: Exchanging edges in matching.

Note that by Claim 7 and by setting weights as above, we ensured that the solution will contain as many w-incident edges as possible. Furthermore, because of the value of $s(w)$ (and, consequently, because of the number of vertices corresponding to w), there can be no more than k'' such edges. As a result, the solution will contain exactly k'' w-incident edges if and only if it is possible for such values of k' and k''. We do not require that there were exactly k' other edges in the matching, but rather set a lower bound, because we can simply ignore excessive edges that are not w-incident (that is: excessive voters that move to the parties not supporting w). Naturally, we do not a'priori know what values of k' and k'' to use. However, there are only polynomially many combinations (we take consecutive integers from $|P_1|$ to 0 as the value of k, and for each of them there are only linearly many decompositions to a sum of two nonnegative integers) and we can try them all. If we search for the value of k in decreasing order (starting with $|P_1|$), then we stop the procedure as soon as we find the first valid solution (thus getting the largest number of voters we can move to other parties). □

On the other hand, for $k \geq 3$, the problem is NP-complete.

THEOREM 8. *k-approval-optimistic-voter-losing, where $k \geq 3$, is* NP-*complete.*

PROOF. The problem is clearly in NP—we can just guess to which parties we should move the voters. To prove NP-hardness of the problem, we construct a reduction from X3C.

Let $B = \{b_1, b_2, \ldots, b_{3t}\}$, $t > 1$, be a set of elements and let $\mathscr{S} = \{S_1, S_2, \ldots, S_m\}$ be a family of three-element subsets of B. Our goal is to construct an instance of k-approval-optimistic-voter-losing problem (defined by an instance of election with parties $E = (C, V, \mathscr{P})$, candidate w and a threshold q), $k \geq 3$, in which there exists a solution if and only if there exists a set $A \subseteq \{1, 2, \ldots, m\}$ such that $|A| = t$, $\bigcup_{i \in A} S_i = B$.

Let us begin with a 3-approval case. We define an election with $3t+3$ candidates: $w, d_1, d_2, c_1, c_2, \ldots, c_{3t}$ and $m+1$ parties such that: (1) The leading party's preference order is $w > d_1 > d_2 > \cdots$ and the party has $t+1$ voters. (2) The other parties have preference orders with the first three candidates representing corresponding 3-sets of the $\mathscr{S}$ family and no members. We ask whether it is possible to keep candidate w winning after moving at least t voters ($q = t$). It is easy to notice that the only way of keeping candidate w winning is to find parties that correspond to the solution of the initial X3C problem and to move one voter to each of them. Thus each candidate would have exactly one point. Otherwise, some candidates would have more than one point, which means that they would beat w. It is clear that the whole reduction works in polynomial time. We can extend this construction to the case of $k > 3$ by adding $k-3$ additional 'dummy' candidates to each party's preference order (different dummies for each party so that no candidate is supported by more than one party). ☐

The optimistic variant of voter losing for k-veto is easy.

THEOREM 9. *k-veto-optimistic-voter-losing is in* P.

PROOF. Let us fix a positive integer k. We will give a polynomial-time algorithm for k-veto-optimistic-voter-losing problem. Our input contains an election $E = (C, V, \mathscr{P})$ with parties, where C is the candidate set, V is a collection of n voters, and $\mathscr{P}$ is the set of t disciplined parties, $\mathscr{P} = \{P_1, \ldots, P_t\}$. Our winning party is P_1 and its top-preferred candidate is w (who, by assumption, is also among the winners of the election). Our goal is to compute the maximum number of voters that can move from P_1 to other parties without preventing w from being a winner.

Let $m+k+1$ be the number of candidates in C. We assume that $m \geq k$ (otherwise we can solve the problem using Theorem 2). We rename the candidates so that $C = \{w, a_1, \ldots, a_m, d_1, \ldots, d_k\}$, where w is the top candidate in party P_1, $a_1, \ldots, a_m$ are the candidates the party P_1 approves of (in addtion to w), and $d_1, \ldots, d_k$ are the candidates that party P_1 ranks on bottom k positions. We set $A = \{a_1, \ldots, a_m\}$ and $D = \{d_1, \ldots, d_k\}$.

Let $\mathscr{P}_w$ be a subset of $\mathscr{P}$ that includes those parties (other than P_1) that approve of w. We claim that if $\mathscr{P}_w$ is empty, then it is impossible to move any voters from P_1 to other parties without preventing w from being a winner.

First, if $\mathscr{P} = \{P_1\}$, then clearly it is impossible to move any voters from P_1 because there is no party where the voters could go. Second, if P_1 is the only party that approves of w, then all the other parties must be initially empty. For the sake of contradiction let us assume otherwise. Let $n_1 > 0$ be the number of P_1 party members and let $n_i > 0$ be the number of party members of some non-empty party P_i. Since $m \geq k$, there must be some candidate c that is approved both by P_1 and by P_i. However, this c has score at least $n_1 + n_i$ which is higher than the score of w (equal to n_1; by assumption, w gets points from party P_1 only). This contradicts the assumption that w is a winner. The same argument shows that if any voter moved from P_1 to some other party, then w would not be a winner anymore.

Thus let us focus on the case where $\mathscr{P}_w$ is nonempty. Through routine calculation, the reader can verify that it suffices to consider the case where voters from P_1 move to parties in $\mathscr{P}_w$. The reason is the following: When a voter moves from P_1 to a party in $\mathscr{P}_w$ then: (a) the score of w does not change, (b) for each candidate $a \in A$, the difference between the score of w and a either stays the same or increases (so w's advantage increases), and (c) for each candidate $d \in D$, the difference between the score of w and d either stays the same or decreases. Thus when voters move from P_1 to parties in $\mathscr{P}_w$ we can disregard candidates in A; their score was originally not greater than that of w and such moves cannot change it. On the other hand, the reader can verify that for each candidate $d \in D$ and each two parties $P' \in \mathscr{P}_w$ and $P'' \in \mathscr{P} - (\mathscr{P}_w \cup \{P_1\})$, the difference between the score of w and d after a voter moves from P_1 to P' is no less than if this voter moved to P''.

This means that we can focus entirely on the voters moving from P_1 to the parties in $\mathscr{P}_w$. Further, it suffices to focus on the scores of candidates in D. As a result, we can describe each party by the subset of those candidates in D that it vetos. If two parties veto the same subsets of candidates in D, then moving a voter from P_1 to either of these parties has the same effect on the score of candidates in D. Thus we attach to each party in $\mathscr{P}_w$ a signature, the subset of candidates in D that it vetos. For each signature that occurs in $\mathscr{P}_w$ we delete all parties except for one with this signature (arbitrarily chosen). Let $\mathscr{P}'_w$ be the collection of parties that we obtain in effect.

Clearly, $\mathscr{P}'_w$ contains at most 2^k parties. If P_1 contains n_1 voters, then there are at most $O(n_1^{2^k})$ ways of moving (some of) them to parties in $\mathscr{P}'_w$. Since k is a constant, we can try all these possibilities in polynomial time and pick the one that moves the maximum number of voters away from P_1. This completes the proof. ☐

The running time of the above algorithm may look somewhat disappointing. After all, even for fairly small values of k, $\Theta(n^{2^k})$ is a prohibitive time complexity. However, in general superpolynomial dependence of the algorithm's running time on k is unavoidable (unless P = NP). This is so because t-approval-optimistic-voter-losing is NP-complete for $t \geq 3$, and for m candidates, $(m-t)$-veto is exactly t-approval.

Borda. The optimistic variant turns out to be computationally hard for Borda's rule. We prove it by a reduction from the X3C problem. This time, however, the construction is not straightforward.

THEOREM 10. *Borda-optimistic-voter-losing is* NP*-complete.*

PROOF. The problem is clearly in NP—we can just guess to which parties we should move the voters. To prove NP-hardness of the problem, we construct a reduction from X3C. Let $B = \{b_1, b_2, \ldots, b_{3t}\}$, $t > 1$, be a set of elements and let $\mathscr{S} = \{S_1, S_2, \ldots, S_m\}$ be a family of three-element subsets of B. Our goal is to construct an instance of Borda-optimistic-voter-losing problem (define an election with parties $E = (C, V, \mathscr{P})$, winner w and threshold q), in which there exists a solution if and only if there exists a set $A \subseteq \{1, 2, \ldots, m\}$ such that $|A| = t$, $\bigcup_{i \in A} S_i = B$. We define an election E with $m+1$ parties $\mathscr{P} = \{P_0, P_1, \ldots, P_m\}$, where P_0 is the leading party and $P_1, \ldots, P_m$ correspond to the 3-sets from $\mathscr{S}$. The election has $3t$ 'real' candidates $b_1, b_2, \ldots, b_{3t}$ corresponding to the elements of B, one winner w and $m \cdot (t-1) \cdot 3t$ 'dummy' candidates $d_i^j(k)$, where $i = 1, \ldots, 3t$, $j = 1, \ldots, t-1$, $k = 1, \ldots, m$. The preference lists of parties are as follows:

1. Party P_0 (the leading one) has preference order $w > b_1 > b_2 > \ldots > b_{3t} > \ldots$

2. Every party P_k ($k = 1, \ldots, m$) has preference order defined as

follows. We take consecutive candidates from b_{3t} to b_1 and for each of them we repeat the procedure described below.

If the selected candidate b_i corresponds to an element of subset S_k from family $\mathscr{S}$, then we put him on place no. $(3t-i)\cdot(t-1)+1$ of the preference list, and we put $t-2$ dummy candidates $d_i^2(k), d_i^3(k), \ldots, d_i^{t-1}(k)$ just below him. Moreover, we put one dummy candidate $d_i^1(k)$ on place no. $3t\cdot(t-1)+i+1$. Otherwise, we put there $t-1$ dummy candidates $d_i^1(k), d_i^2(k), \ldots, d_i^{t-1}(k)$ (starting from the same place as we would have put candidate b_i in the previous case) and we put candidate b_i on place no. $3t\cdot(t-1)+i+1$. Finally, we put candidate w on place no. $3t\cdot(t-1)+1$.

After that, we have the first $3t\cdot t+1$ places of the list occupied. The other places may be taken by the rest of the dummy candidates in any order.

Assuming that $S_k = \{b_p, b_q, b_r\}$, the preference order of the corresponding party P_k is $Block_{3t}(k) > Block_{3t-1}(k) > \ldots > MemberBlock_r(k) > \ldots > MemberBlock_q(k) > \ldots > MemberBlock_p(k) > \ldots > Block_1(k) > w > b_1 > \ldots > b_{p-1} > d_p^1(k) > b_{p+1} > \ldots > b_{q-1} > d_q^1(k) > b_{q+1} > \ldots > b_{r-1} > d_r^1(k) > b_{r+1} > \ldots > b_{3t}$, where $Block_i(k)$ is given by $d_i^1(k) > d_i^2(k) > \ldots > d_i^{t-1}(k)$ and $MemberBlock_i(k)$ corresponds to $b_i > d_i^2(k) > \ldots > d_i^{t-1}(k)$.

The leading party has t voters and all of the other parties have no voters. We ask whether it is possible to keep candidate w winning after moving at least t (or rather exactly t, since there are only t voters in the election) voters (we set the threshold $q = t$).

First of all, it is clear that—having the election instance defined as above—candidate w initially wins the election: Initially, the winner w gets $\text{score}_E(w) = t\cdot(|C|-1)$ points, and any of the other 'real' candidates b_i for $i = 1,\ldots,3t$ gets $\text{score}_E(b_i) = t\cdot(|C|-i-1) < \text{score}_E(w)$, where $|C|$ denotes the number of candidates in the election. Obviously, any of the dummy candidates gets even fewer points.

Second, note that if there exists a solution to the original X3C problem, then the answer to the Borda-optimistic-voter-losing problem is also positive: The preference lists are organized in such a way that—after moving voters in accordance with the X3C solution (one voter for each party representing a subset from the cover)—candidate w and all of the candidates $b_1,\ldots,b_{3t}$ will have an equal number of points (recall that E' denotes the transformed election that we get from E after moving all voters according to the description provided in this proof):
After moving t voters according to the X3C solution, the candidate w gets $\text{score}_{E'}(w) = t\cdot(|C|-3t\cdot(t-1)-1)$ points and any of the other candidates b_i, $i = 1,\ldots,3t$ gets $\text{score}_{E'}(b_i) = |C|-(3t-i)\cdot(t-1)-1+(t-1)\cdot(|C|-3t\cdot(t-1)-i-1) = t\cdot(|C|-3t\cdot(t-1)-1) = \text{score}_{E'}(w)$, where the term $|C|-(3t-i)\cdot(t-1)-1$ comes from the voter moved to the party corresponding to the set from X3C solution that covers element b_i, and the term $(t-1)\cdot(|C|-3t\cdot(t-1)-i-1)$ comes from voters moved to the other $t-1$ parties. Note that the last result (the number of points scored by candidate b_i) is independent of the index i, which means that every candidate gets exactly the same score. Furthermore, in such case none of the dummy candidates will have a chance of winning the election. Dummy candidates can score at most $|C|-1+(t-1)\cdot(|C|-3t\cdot t-2) = t\cdot|C|-3t\cdot t\cdot(t-1)-2(t-1)-1 < \text{score}_{E'}(w)$. This is the most optimistic case, where one of the dummy candidates is top-ranked by one of the nonempty parties (thus getting $|C|-1$ points), and all other nonempty parties rank him on the highest possible place (i.e., on place number $3t\cdot t+2$). The last inequality holds because we assumed that $t > 1$ (recall the assumption about the size of X3C input set B).

Conversely, if there exists a solution to the optimistic-voter-losing problem for a scoring protocol, then there must exist a solution to the corresponding X3C problem. Let us suppose that exactly t voters have moved to other (non-leading) parties. It is clear that in such a case, candidate w (the initial winner of the election) would have exactly $t\cdot(|C|-3t\cdot(t-1)-1)$ points, regardless of which parties had the voters moved to (since all parties, except for the leading party P_0, rank w on the same place). Let us fix some arbitrary candidate $c \in C\setminus\{w\}$. Let $\alpha(c)$ and $\beta(c)$ denote the sets of all those parties, which rank candidate c above and below candidate w, respectively. If a party is a member of the set $\alpha(c)$, then we will say that it *covers* candidate c. Each of the candidates b_i, $i = 1,\ldots,3t$, gets $|C|-(3t-i)\cdot(t-1)-1$ points for each voter that moved to one of the parties $\alpha(b_i)$, and $|C|-3t\cdot(t-1)-i-1$ points for each voter that moved to one of the parties $\beta(b_i)$, so switching his choice between those two options changes his overall score by $(|C|-(3t-i)\cdot(t-1)-1)-(|C|-3t\cdot(t-1)-i-1) = t\cdot i$. Supposing that for some $i = 1,\ldots,3t$ all t voters have moved only to parties from $\beta(b_i)$, candidate b_i would have a total of $\text{score}_{E'}(b_i) = t\cdot(|C|-3t\cdot(t-1)-i-1)$, which is such that $\text{score}_{E'}(w)-\text{score}_{E'}(b_i) = t\cdot i$. Consequently, each candidate may be covered at most once—otherwise he or she will overtake w, which stays in opposition to the conditions of Borda-optimistic-voter-losing problem. Furthermore, it means that we can move at most one voter to each party. On the other hand, each party covers exactly three candidates. Finally, mind that we move exactly t voters, which implies—in conjunction with the statement above—that we choose exactly t parties. Taking all of the statements above into consideration we can clearly see that this is precisely the definition of the X3C problem.

It remains to note that the reduction is polynomial-time. □

5. CONDORCET CONSISTENT RULES

We now present our complexity results regarding two Condorcet consistent rules (namely: Copeland and Maximim).

Maximin. The pessimistic variant for maximin is difficult.

THEOREM 11. *Maximin-pessimistic-voter-losing is* coNP-*complete.*

PROOF. It is clear that the problem is in coNP and thus we focus on the proof of its coNP-hardness. We do so via a reduction from the complement of X3C (that is, the problem that is identical to X3C, except that if the answer for a given X3C instance is "yes" then in the complement the answer is "no", and the other way round). Let $I = (B, \mathscr{S})$ be an instance of X3C, where $B = \{b_1,\ldots,b_{3k}\}$ and $\mathscr{S} = \{S_1,\ldots,S_n\}$ is a family of three-eleemt subsets of B. We build the following election $E = (C, V, \mathscr{P})$ with disciplined parties. The candidate set is $B \cup \{a,b,c,d,w\}$. The set of parties is $\mathscr{P} = \{P_1, P_2, R_1, \ldots, R_6, T_1, \ldots T_4, Q_1, \ldots, Q_n\}$ with the following preference orders and member counts (we set $N = 4k$, $X = 5k$, $Z = 4k$, $Y = k+1$ and $W = k-1$):

1. Both P_1 and P_2 have N members each. P_1 has preference order $w > C-\{w,d\} > d$ and P_2 has preference order $w > \overleftarrow{C-\{w,d\}} > d$.
2. Parties R_1 and R_2 have X members each. R_1 has preference order $a > b > d > c > w > B$ and R_2 has preference order $\overleftarrow{B} > c > w > d > b > a$.
3. Parties R_3 and R_4 have Z members each. R_3 has preference order $w > b > d > a > c > B$ and R_4 has preference order $\overleftarrow{B} > a > c > d > b > w$.

	w	a	b	c	d	b_i	b_j	score
w	–	$8k$	$8k$	$-2k$	$6k-2$	$8k$	$8k$	$-2k$
a	$-8k$	–		$8k$	$6k-2$			$-8k$
b	$-8k$		–		$6k-2$			$-8k$
c	$2k$	$-8k$		–	$6k-2$			$-8k$
d	$-6k+2$	$-6k+2$	$-6k+2$	$-6k+2$	–	$-4k$	$-4k$	$-6k+2$
b_i	$-8k$				$4k$	–		$-8k$
b_j	$-8k$				$4k$		–	$-8k$

Table 1: Function $M_E(\cdot,\cdot)$ from the proof of Theorem 11. Candidates b_i and b_j are two arbitrarily chosen, distinct, members of B. Empty cells indicate that the value of the function for given two candidates is 0.

4. Parties R_5 and R_6 have W members each. R_5 has preference order $a > d > B > b > c > w$ and R_6 has preference order $w > c > b > d > \overleftarrow{B} > a$.
5. Parties T_1, T_2, T_3, and T_4 have Y members each, and have the following preference orders:
   (a) $T_1 : a > d > b > c > w > B$,
   (b) $T_2 : b > d > a > w > c > \overleftarrow{B}$,
   (c) $T_3 : c > d > w > a > b > \overleftarrow{B}$, and
   (d) $T_4 : \overleftarrow{B} > w > d > c > b > a$.
6. For each i, $1 \le i \le n$, party Q_i has 0 members and has preference order $B - S_i > d > S_i > a > b > c > w$.

Table 1 shows the values of function $M_E(\cdot,\cdot)$ and Maximin scores of the candidates prior to any voter movement in E. We claim that there is an exact cover of B with sets from $\mathscr{S}$ if and only if there is a way to move k voters from P_1 to other parties so that w ceases to be a winner.

Let us assume that there is no exact cover of B with sets from $\mathscr{S}$. After we move k voters from P_1 to other parties, the score of w decreases to no less than $-4k$. The score of the candidates in $B \cup \{a,b,c\}$ cannot increase to more than $-6k$ so neither of them can endanger w's victory. The only candidate whose score could possibly exceed $-4k$ is d. However, if even one voter v of our k moving voters go from P_1 to some party in $\{P_2, R_1, \dots, R_6, T_1, \dots, T_4\}$ then the score of d cannot go beyond $-4k$. The reason is that this v will move to a party that prefers some candidate $x \in \{a,b,c,w\}$ to d, and thus in the election E' resulting from the move $M_{E'}(d,x) \le -4k$, and d will not defeat w. Yet, if all the k voters that leave party P_1 go to parties in $\{Q_1, \dots, Q_n\}$ then, since there is no cover of B with sets from $\mathscr{S}$, there will be at least one candidate $b_i \in B$ such that in the election E'' resulting from the move, $M_{E''}(d,b_i) = -4k$ and, again, d will not defeat w.

On the other hand, if there is an exact cover of B with sets from $\mathscr{S}$, then if k voters move from P_1 to the parties corresponding to the cover, then d will have score $-4k+2$ and w will have score $-4k$. That is, there is a way to move k voters that prevents w from being a winner. □

THEOREM 12. *For Maximin, optimistic-voter-losing is* NP-*complete.*

Copeland. The results for the case of Copeland are the same as for Maximin and we omit the proof for the pessimistic case due to space constraints; it is similar in spirit to the proof used for Maximin. The key trick is to use McGarvey's theorem [15] in a way that allows us to build an election with necessary structure, but which prevents voters to move from P_1 to the parties that we use in the application of McGarvey's theorem. This can be done using candidates a, b, c, and d similarly as in the proof of Theorem 11. The proof for the optimistic case relies on a similarity to set covering.

THEOREM 13. *For each rational α, $0 \le \alpha \le 1$, Copeland$^\alpha$-pessimistic-voter-losing is* coNP-*complete.*

THEOREM 14. *For each rational α, $0 \le \alpha \le 1$, Copeland$^\alpha$-optimistic-voter-losing is* NP-*complete.*

PROOF. We will give a proof for the case of $\alpha = 0$ (we do so due to space constraints; a proof that covers all values of α jointly is somewhat longer). It is clear that the problem is in NP and we will show its NP-hardness via a reduction from X3C. Let $I = (B, \mathscr{S})$ be an instance of X3C where $B = \{b_1, \dots, b_{3k}\}$ and $\mathscr{S} = \{S_1, \dots, S_n\}$ is a family of three-element subsets of B.

We form two sets, $B' = \{b'_1, \dots, b'_{3k}\}$ and $B'' = \{b''_1, \dots, b''_{3k}\}$. Further, for each set $S_i \in \mathscr{S}$, we form a set $R_i = \{b'_j \mid b_j \in S_i\} \cup \{b''_j \mid b_j \in S_i\}$. We construct an election $E = (C, V, \mathscr{P})$ with parties, where $C = B' \cup B'' \cup \{w, c\}$, V is a collection of $4k$ voters, and $\mathscr{P} = \{P_1, P_2, Q_1, \dots, Q_n\}$ is a set of parties.

Party P_1 has $2k+2$ members and preference order $w > c > B' > B''$. Party P_2 has $2k-2$ members and preference order $c > \overleftarrow{B''} > \overleftarrow{B'} > w$. For each i, $1 \le i \le n$, party Q_i has 0 members and preference order $R_i > w > c > (B' \cup B'') - R_i$. Prior to the movement of any voters, w wins head-to-head contests with all the other candidates (each by 4 votes) and has score $6k+1$. Candidate c loses his head-to-head contest to c by 4 votes, but wins every head-to-head contest with candidates in $B' \cup B''$ by $4k$ votes. Altogether, c has score $6k$ (and this cannot change after the movement of at most k voters from P_1). Each candidate in $B' \cup B''$ loses both to w and to c, and thus has score at most $6k-1$.

We claim that it is possible to move k voters away from party P_1 if and only if there is an exact cover of B with sets from $\mathscr{S}$. Let us assume that there is such a cover. If we move voters from P_1 to the parties in $\{Q_1, \dots, Q_n\}$ that correspond to the sets from the cover, then the score of w is still $6k+1$ and, as a result, w is still a winner.

On the other hand, let us assume that there is no exact cover. If we move k voters from P_1 to the parties in $\{Q_1, \dots, Q_n\}$ then, since there is no exact cover of B, there will be at least two candidates, $b'_i \in B'$ and $b''_i \in B''$, with whom w will either tie or lose. In effect w will have score at most $6k-1$ and since c will still have score $6k$, w will not be a winner any more. Alternatively, if k voters will move from P_1 to other parties and at least one of them will move to P_2, then clearly the score of w will be at most $6k-5$ and so w will have fewer points than c. Since the reduction can be computed in polynomial time, the proof is complete. □

6. RELATED WORK

There is a long collection of papers that study computational complexity of affecting the results of elections, and we point the reader to the survey papers of Faliszewski, Hemaspaandra, and Hemaspaandra [9] and of Brandt, Conitzer, and Endriss [3].

Our work is perhaps closest to a very recent paper of Baumeister et al. [1], where the authors asks the following question: Given an election, is it possible to set the weights of the voters so that a particular candidate becomes a winner? Our work differs in important technical details: First, in our case the total weight of the parties (that is, the total number of members of parliament) is fixed and each member has to belong to some party. Second, we have initial "weights" of the parties and voters can leave only the specified leading party. (In terms of Baumeister et al., for one voter we can only decrease the weight, and for the others we can only increase the weights, and the total weight has to stay the same.) Further, in our case voters can go from the leading party to any of the other parties, whereas Baumeister et al. have two groups of voters, those whose weights are fixed and those whose weights may change. Thus our complexity results differ from theirs.

Our work is also related to Faliszewski, Hemaspaandra, and Hemaspaandra's [10] work on multimode control, where the authors study questions such as: If I am allowed to both add and delete some voters, then what is the complexity of ensuring that a given candidate is a winner? Our problem can be viewed as a variant of simultaneous adding and deleting voters (we delete voters from the leading party and add them to the other parties).

Our problems resemble also the problem of bribery [8] and margin of victory (destructive bribery) [19]. One could also use the very recent result of Xia [20] to provide asymptotic analysis of our problems in terms of the expected number of voters that have to leave the leading party in order to change the winner. Finally, our work is close in spirit to the studies of possible and necessary winners [12, 13, 17, 21], though different on technical side.

7. CONCLUSIONS

The paper provides a study of party changing problem in optimistic and pessimistic variant for the most common scoring protocols (namely, plurality, veto, k-approval, k-veto, and Borda) and for Cordorcet consistent rules (namely, Copeland and Maximin). Table 2 shows a summary of our results. As opposed to the studies of manipulation, bribery, and control in elections (see [3, 9]), here NP-completeness should not be taken to mean that members of the leading party will not attempt to threaten the result of an election by moving to other parties. Rather, where we have polynomial-time algorithms for both our problems, we should expect the party members attempting to move and the party leader trying to keep them, to make rational calculations regarding their situation. Where our results show NP-completeness (or coNP-completeness), these agents' calculations would, by necessity, be only approximate or take much more of their resources.

Table 2: The complexity of the party-change problem

Election system	pessimistic	optimistic
plurality	P	P
veto	P	P
2-approval	P	P
k-approval (k≥ 3)	P	NP-com
k-veto	P	P
Borda	P	NP-com
maximin	coNP-com	NP-com
Copeland$^{\alpha}$ (for all α)	coNP-com	NP-com

We plan to investigate complexity results for this problems also in scenarios where we have probabilistic information about the parties where the voters will move, in the spirit of probabilistic study of control problems initiated by Wojtas and Faliszewski [18].

Acknowledgments. The authors were supported in part by AGH Univ. grant 11.11.120.865, by the Foundation for Polish Science's Homing/Powroty program, by Poland's National Science Center grant DEC-2011/03/B/ST6/01393, and by the Italian MIUR PRIN 20089M932N project "Innovative and multi-disciplinary approaches for constraint and preference reasoning".

8. REFERENCES

[1] D. Baumeister, M. Roos, J. Rothe, L. Schend, and L. Xia. The possible winner problem with uncertain weights. In *Proc. of ECAI-12*, pages 133–138, Aug. 2012.

[2] S. Brams and P. Fishburn. Voting procedures. In K. Arrow, A. Sen, and K. Suzumura, editors, *Handbook of Social Choice and Welfare*, pages 173–236. Elsevier, 2002.

[3] F. Brandt, V. Conitzer, and U. Endriss. Computational social choice. In G. Weiß, editor, *Multiagent Systems*. MIT Press, 2012. Forthcoming.

[4] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th International World Wide Web Conference*, pages 613–622. ACM Press, Mar. 2001.

[5] E. Ephrati and J. Rosenschein. The Clarke Tax as a consensus mechanism among automated agents. In *Proc. of AAAI-91*, pages 173–178, July 1991.

[6] E. Ephrati and J. Rosenschein. Multi-agent planning as a dynamic search for social consensus. In *Proc. of IJCAI-93*, pages 423–429. Morgan Kaufmann, 1993.

[7] R. Fagin, R. Kumar, and D. Sivakumar. Efficient similarity search and classification via rank aggregation. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, pages 301–312. ACM Press, June 2003.

[8] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra. How hard is bribery in elections? *JAIR*, 35:485–532, 2009.

[9] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra. Using complexity to protect elections. *Commun. ACM*, 53(11):74–82, 2010.

[10] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra. Multimode control attacks on elections. *JAIR*, 40:305–351, 2011.

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

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

[13] J. Lang, M. S. Pini, F. Rossi, D. Salvagnin, K. B. Venable, and T. Walsh. Winner determination in voting trees with incomplete preferences and weighted votes. *Autonomous Agents and Multiagent Systems*, 25(1):130–157, 2012.

[14] L. Lovasz and M. Plummer. *Matching Theory*. 1986.

[15] D. McGarvey. A theorem on the construction of voting paradoxes. *Econometrica*, 21(4):608–610, 1953.

[16] D. Pennock, E. Horvitz, and C. Giles. Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering. In *Proc. of AAAI-00*, pages 729–734, July/August 2000.

[17] M. S. Pini, F. Rossi, K. B. Venable, and T. Walsh. Incompleteness and incomparability in preference aggregation: Complexity results. *Artificial Intelligence*, 175(7-8):1272–1289, 2011.

[18] K. Wojtas and P. Faliszewski. Possible winners in noisy elections. In *Proc. of AAAI-12*, pages 1499–1505, July 2012.

[19] L. Xia. Computing the margin of victory for various voting rules. In *Proc. of EC-12*, pages 982–999, June 2012.

[20] L. Xia. How many vote operations are needed to manipulate a voting system? In *Proc. of COMSOC-12*, 2012.

[21] L. Xia and V. Conitzer. Determining possible and necessary winners given partial orders. *JAIR*, 41:25–67, 2011.