Possible Winners when New Candidates are Added: the Case of Scoring Rules

Yann Chevaleyre and **Jérôme Lang** and **Nicolas Maudet** and **Jérôme Monnot**
LAMSADE
Université Paris-Dauphine
France
{yann.chevaleyre,jerome.lang,nicolas.maudet,jerome.monnot}@dauphine.fr

Abstract

In some voting situations, some new candidates may show up in the course of the process. In this case, we may want to determine which of the initial candidates are possible winners, given that a fixed number k of new candidates will be added. Focusing on scoring rules, we give complexity results for the above possible winner problem.

Introduction

In many real-life collective decision making situations, the set of candidates (or alternatives) may vary while the voting process goes on, and may change at any time before the decision is final: some new candidates may join, whereas some others may withdraw. This, of course, does not apply to situations where the vote takes place in a very narrow period of time (such as, typically, political elections in most countries), and the addition of new candidates during the process does not apply either to situations where the law forbids new candidates to be introduced after the vote has started (which, again, is the case for most political elections). However, there are quite many practical situations where this situation does happen, especially contexts where votes are sent by email during an extended period of time. This is typically the case when making a decision about the date and time of a meeting. In the course of the process, we may learn that the room is taken at a given time slot, making this time slot no longer a candidate. The opposite case also occurs frequently; we thought the room was taken on a given date and then we learn that it has become available, making this time slot a new candidate.

The paper focuses on candidate addition only. More precisely, the class of situations we consider is the following. A set of voters have expressed their votes about a set of (initial) candidates. Then some new candidates declare. The winner will ultimately be determined using some given voting rule. In this class of situations, an important questions arises: *who among the initial candidates can still be a winner once the voters' preferences about all candidates are known?*

This question is strongly related to several streams of work from the recent literature on computational social choice, especially the problem of determining whether the vote elicitation process is terminated (Conitzer and Sandholm 2002b; Walsh 2008); the possible winner problem, and more generally the problem of applying a voting rule to incomplete (Konczak and Lang 2005; Pini et al. 2007; Xia and Conitzer 2008; Betzler and Dorn 2009; Betzler, Hemmann, and Niedermeier 2009) or uncertain (Hazon et al. 2009) preferences; and finally, to the control of a voting rules by the chair via adding candidates —we shall come back on the latter problem later on.

Clearly, the situations where new voters are added is a case of voting under incomplete preferences, where incompleteness is of a very specific type: the set of candidates is partitioned in two groups (the initial and the new candidates), and the incomplete preferences consist of complete rankings on the initial candidates. This class of situations is somehow dual of a class of situations that has been considered more often, namely, when the set of *voters* is partitioned in two groups: those voters who have already voted, and those who haven't expressed their votes yet. The latter class of situations, while being a subclass of voting under incomplete preferences, has been more specifically studied as a *coalitional manipulation problem* (Conitzer and Sandholm 2002a; Xia et al. 2009), where the problem is to determine whether it is possible for the voters who haven't voted yet to make a given candidate win. Varying sets of voters have also been studied in the context of compiling the votes of a subelectorate (Chevaleyre et al. 2009): there, one is interested in synthesizing a set of initial votes, while still being able to compute the outcome once the remaining voters have expressed their votes. Finally, the possible winner problem via candidate addition is highly related to manipulation by candidate cloning. The main difference is that cloning requires a candidate and its clones to be contiguous. The complexity of this problem is considered in Elkind et al. (Elkind, Faliszewski, and Slinko 2010). Although the proposed model allows for the possibility of having a bounded number of new clones, most of their results focus on the case of unboundedly many clones, which also differs from our case.

The layout of the paper is as follows. We start by recalling the necessary background on voting. Then we state the problem formally, by defining voting situations where candidates may be added in the course of the process; at this point we will discuss the relationship to the control of an election by the chair via adding candidates. Then we consider the possible and necessary winner problem from a complexity point of view. After stating the problem formally, we establish a collection of results showing that even for seemingly rather similar voting rules belonging to the class of scoring rules, various levels of complexity are encountered. Finally, we mention further research directions.

Background

Let C be a finite set of *candidates*, and N a finite set of *voters*. Let $p = |C|$ and $n = |N|$. A C-*vote* (called more simply a vote when this is not ambiguous) is a linear order over C. We sometimes denote votes in the following way: $a \succ b \succ c$ is denoted by abc, etc. A C-*profile* is a collection $P = \langle V_1, \ldots, V_n \rangle$ of C-votes. Let $\mathcal{P}_C$ be the set of all C-votes and therefore $\mathcal{P}_C^n$ be the set of all n-voter C-profiles.

A voting rule on C is a function r from $\mathcal{P}_C^n$ to C. As the usual definition of most voting rules does not exclude the possibility of ties, we assume these ties are broken by a fixed priority order on candidates.

For $P \in \mathcal{P}_C^n$ and $x, x' \in C$, let $n(P, i, x)$ be the number of votes in P ranking x in position i, $ntop(P, x) = n(P, 1, x)$ the number of votes in P ranking x first, and $N_P(x, x')$ the number of votes in P ranking x above x'. Let $\vec{s} = \langle s_1, \ldots, s_p \rangle$ be a vector of integers such that $s_1 \geq \ldots \geq s_p$ and $s_1 > s_p$. The scoring rule $r_{\vec{s}}(P)$ induced by $\vec{s}$ elects the candidate maximizing $score_{\vec{s}}(x, P) = \sum_{i=1}^{p} s_i.n(P, i, x)$. The *plurality* rule r_P is the scoring rule corresponding to the vector $\langle 1, 0, \ldots, 0 \rangle$. The *Borda* rule r_B is the scoring rule corresponding to the vector $\langle p-1, p-2, \ldots, 0 \rangle$. The *veto* rule r_V is the scoring rule corresponding to the vector $\langle 1, \ldots, 1, 0 \rangle$. If K is a fixed integer then K-*approval*, r_K, is the scoring rule corresponding to the vector $\langle 1, \ldots, 1, 0, \ldots, 0 \rangle$ – with K 1's and $p - K$ 0's.

We now define formally situations where new candidates are added.

Definition 1 *A* voting situation with varying candidates *is a 4-uple* $\Sigma = \langle N, X, P_X, k \rangle$ *where* N *is a set of voters (with* $|N| = n$*),* X *a set of candidates,* $P_X = (V_1, \ldots, V_n)$ *a* n*-voter* X*-profile, and* k *is a positive integer.*

X denotes the set of initial candidates, P_X the initial profile, and k the number of new candidates. Nothing is known a priori about the voters' preferences relatively to the new candidates, henceforth their identity is irrelevant and only their number counts.

Because the number of candidates is not the same before and after the new candidates come in, we have to consider families of voting rules (for a varying number of candidates) rather than voting rules for a fixed number of candidates. While it is true that for many usual voting rules there is an obvious way of having them defined for a varying number of candidates, this is not the case for all of them, especially scoring rules other than plurality, Borda and veto. We shall therefore consider *collections of voting rules*, parameterized by the number of candidates. We slightly abuse notation and denote these collections of voting rules by r. Again with a slight abuse of notation, we often write $r(P)$ instead of $r_p(P)$. The complexity results we give in this paper bear on such collections of voting rules, where the number of candidates is variable.

If P is a C-profile and $C' \subseteq C$, then the projection of P on C', denoted by $P^{\downarrow C'}$, is obtained by deleting all candidates in $C \setminus C'$ in each of the votes of P, and leaving unchanged the ranking on the candidates of C'. For instance, let us take $P = \langle abcd, dcab \rangle$ then $P^{\downarrow \{a,b\}} = \langle ab, ab \rangle$ and $P^{\downarrow \{a,b,c\}} = \langle abc, cab \rangle$. In all situations, the set of initial candidates is denoted by $X = \{x_1, \ldots, x_p\}$, the set of the k new candidates is denoted by $Y = \{y_1, \ldots, y_k\}$. If P_X is a X-profile and P a $X \cup Y$-profile, then we say that P extends P_X if the projection of P on the candidates in X is exactly P_X. For instance, let $X = \{x_1, x_2, x_3\}$, $Y = \{y_1, y_2\}$; the profile $P = \langle x_1y_1x_2y_2x_3, y_1y_2x_1x_2x_3, x_3x_2y_2y_1x_1 \rangle$ extends the X-profile $P = \langle x_1x_2x_3, x_1x_2x_3, x_3x_2x_1 \rangle$.

Possible winners when new candidates are added

We recall from (Konczak and Lang 2005) that given a collection $\langle P_1, \ldots, P_n \rangle$ of partial strict orders on C representing some incomplete information about the votes, a candidate x is a possible winner if there is a profile $\langle T_1, \ldots, T_n \rangle$ where each T_i is a ranking on C extending P_i in which x wins. Reformulated in the case where P_i is a ranking of the initial candidates (those in X), we get the following definition:

Definition 2 *Given a voting situation* $\Sigma = \langle N, X, P_X, k \rangle$*, a set of new candidates* Y *(with* $|Y| = k$*), and a collection* r *of voting rules, we say that* $x \in X$ *is a* possible winner with respect to Σ, Y, and r *if there is a* $X \cup Y$*-profile* P *extending* P_X *such that* $r(P) = x$*.*

For all results in the paper, x will be assumed to have the most favorable priority (in case of a tie between x and other candidates, x is the winner). We briefly discuss this issue in the conclusion.

We do not define the notion of *necessary winner* with respect to Σ and r, because except for extremely specific voting rules, there will *never* be a necessary winner when we add k new candidates. Indeed, provided that we have a very weak condition on r which says that if y is ranked first by every voter then it is the winner[1], the new candidates will always be possible winners.

Possible winners with respect to the addition of candidates are reminiscent of constructive control by the chair via adding candidates — this problem first appeared in (Bartholdi, Tovey, and Trick 1992) and was later studied in more depth for many voting rules, see *e.g.*, (Hemaspaandra, Hemaspaandra, and Rothe 2007; Faliszewski, Hemaspaandra, and Hemaspaandra 2009). However, constructive control via adding candidates significantly differs from the problem studied in this paper. In control problems, the chair knows how the voters would vote on the new candidates, and we want to decide whether a given candidate can be made a winner by adding at most k new candidates. Here, we don't

[1]Note however that this property is violated *e.g.* by the veto rule if some $x \in X$ has priority over all the y_i's for tie-breaking, so that the veto rule may have a necessary winner when adding k new candidates.

have the faintest knowledge of how the voters will rank the new candidates.

Now we are in position to consider specific voting rules.

Plurality and veto

Let us start with an example: suppose $X = \{a, b, c\}$, $n = 12$, and the plurality scores in P_X are $a \mapsto 5$, $b \mapsto 4$, $c \mapsto 3$, while the tie-breaking priority is $a > b > c > y$. There is only one new candidate (y). We have that: (1) a is a possible winner (a will win in particular if the top candidate of every voter remains the same); (2) b is a possible winner: to see this, suppose that 2 voters who had ranked a first now rank y first; the new scores are $a \mapsto 3$, $b \mapsto 4$, $c \mapsto 3$, $y \mapsto 2$; and (3) c is not a possible winner: to make her having a higher score than the scores of a and b, we need at least 3 (resp. 2) voters who had ranked a (resp. b) first now rank y first; but this then means that y gets at least 5 votes, while c has only 3. More generally, we have the following result:

Proposition 1 *Let P_X be an n-voter profile on X, and $x \in X$. x is a possible winner for P_X and plurality with respect to the addition of k new candidates if and only if*

$$ntop(P_X, x) \geq \frac{1}{k} . \sum_{x_i \in X} \max(0, ntop(P_X, x_i) - ntop(P_X, x))$$

Proof: Suppose first that the inequality holds. We build the following $(X \cup Y)$-profile P extending P_X:

1. for every candidate x_i such that $ntop(P_X, x_i) > ntop(P_X, x)$ we take $ntop(P_X, x_i) - ntop(P_X, x)$ arbitrary votes ranking x_i on top and place one of the y_j's on top of the vote (and the other y_j's anywhere), subject to condition 2 below.
2. place the y_j's on top of the votes in such a way that no y_j is placed on top of a vote more than $ntop(P_X, x)$ times. This is possible because the inequality is satisfied.
3. in all other votes (those not considered at step 1), place all y_j's anywhere except on top.

We obtain a profile P extending P_X. First, we have $ntop(P, x) = ntop(P_X, x)$, because on all the votes in P_X where x is on top, the new top candidate in the corresponding vote in P is still x, cf. step 3), and all the votes in P_X where x was not on top obviously cannot have x on top in the corresponding vote in P. Second, let $x_i \neq x$. If $ntop(P_X, x_i) \leq ntop(P_X, x)$ then $ntop(P, x_i) = ntop(P_X, x_i)$; and if $ntop(P_X, x_i) > ntop(P_X, x)$ then we have $ntop(P, x_i) = ntop(P_X, x_i) - (ntop(P_X, x_i) - ntop(P_X, x)) = ntop(P_X, x)$. Therefore, the winner for plurality in P is x.
Conversely, if the inequality is not satisfied then, in order x to become the winner in P, the other x_i's must lose globally an amount of $\sum_{x_i \in X} \max(0, ntop(P_X, x_i) - ntop(P_X, x))$ votes; and since $\sum_{x_i \in X} \max(0, ntop(P_X, x_i) - ntop(P_X, x)) > k.ntop(P_X, x)$, for at least one of the y_j's we will have $ntop(P, y_j) > ntop(P, x)$; therefore x cannot be the winner for plurality in P. ■

For veto, this is almost trivial. By placing any of the new candidates below x in every vote of P_X where x is ranked at the bottom position, we obtain a vote P where no one vetoes x. Even if some other candidates $x_i \neq x$ are not vetoed in P_X, the tie will break in favour of x: any candidate is then a possible winner under this assumption.

Therefore, computing possible winners for plurality and veto with respect to candidate addition is polynomial (which we already knew, since possible winners for plurality and veto can be computed in polynomial time (Betzler and Dorn 2009)).

K-approval

For any $x_j \in X$ we denote by $S_K(P_X, x_j)$ the score of x_j for P_X and K-approval (*i.e.* the number of voters who rank x_j among their top K candidates); and by $Z_K(P_X, x_j)$ the number of voters who rank x_j exactly in position K. We start by the case where a single candidate is added.

Proposition 2 *Let $1 \leq K \leq p - 1$, P_X an n-voter profile on X, and $x \in X$. x is a possible winner for P_X and K-approval with respect to the addition of one new candidate if and only if the following two conditions hold:*

1. *for any $x_i \neq x$, if $S_K(P_X, x_i) > S_K(P_X, x)$ then $Z_K(P_X, x_i) \geq S_K(P_X, x_i) - S_K(P_X, x)$.*
2. *$S_K(P_X, x) \geq \sum_{x_i \in X} \max(0, S_K(P_X, x_i) - S_K(P_X, x))$*

Proof: Assume conditions (1) and (2) are satisfied. then we build the following $(X \cup Y)$-profile extending P_X:

- for every x_i such that $S_K(P_X, x_i) > S_K(P_X, x)$, we take $S_K(P_X, x_i) - S_K(P_X, x)$ arbitrary votes who rank x_i in position K in P_X and place y on top. This is possible because condition (1) is satisfied.
- in all other votes (those not considered at step 1), place all y at the bottom.

We obtain a profile P extending P_X. First, we have $S_K(P, x) = S_K(P_X, x)$, because (a) all votes in P_X ranking x in position K are extended in such a way that all y is placed on the bottom position, therefore x gets a point in each of these votes if and only if it got a point in P_X, and (b) all votes ranking x in position other than K in P_X get a point in P if and only if they get a point in P, both in the case y was added on top of the vote and in the case it was added on bottom of the vote. Second, for every x_i such that $S_K(P_X, x_i) > S_K(P_X, x)$, x_i loses exactly $S_k(P_X, x_i) - S_k(P_X, x)$ points when P_X is extended into P, therefore $S_K(P, x_i) = S_K(P_X, x_i) - S_K(P_X, x_i) + S_K(P_X, x) = S_K(P_X, x)$. Third, $S_K(P, y) = \sum_{x_i \in X \setminus \{x_i\}} \max(0, S_K(P_X, x_i) - S_K(P_X, x)) \leq S_K(P_X, x)$ – because of (2) – hence $S_K(P, y) \leq S_K(P, x)$. Therefore, the winner for K-approval in P is x.

Now, assume condition (1) is not satisfied, that is, there is a x_i such that $S_K(P_X, x_i) > S_K(P_X, x)$ and such that $Z_K(P_X, x_i) < S_K(P_X, x_i) - S_K(P_X, x)$. There is no way of having x_i losing more than $Z_K(P_X, x_i)$ points, therefore x will never catch up x_i's advance and is therefore not a possible winner. Finally, assume condition (2) is not satisfied, which means that we have $\sum_{x_i \in X \setminus \{x_i\}} \max(0, S_K(P_X, x_i) - S_K(P_X, x)) > S_K(P_X, x)$. Then, to catch up the advance of the x_i's on x we must add y in one of the top K position in a number of votes exceeding $S_K(P_X, x)$, therefore $S_K(P, y) > S_K(P_X, x) \geq S_K(P, x)$, and therefore x is not a possible winner. ■

Therefore, computing possible winners for $K-$approval with respect to the addition of *one* candidate can be done in polynomial time. However, this result cannot be generalized to K-approval. The following result shows that for 4-approval, 3 new candidates suffice to turn it into an NP-complete problem.

Proposition 3 *Deciding if x^* is a possible winner for 4-approval with respect to the addition of three new candidates, is an* NP*-complete problem.*

Proof: This problem is clearly in NP. The proof is based on a reduction from the 3-dimensional matching problem, denoted by 3-DM. An instance of 3-DM consists of a subset $\mathcal{C} = \{e_1, \ldots, e_m\} \subseteq X \times Y \times Z$ of triples, where X, Y, Z are 3 pairwise disjoint sets of size n with $X = \{x_1, \ldots, x_n\}$, $Y = \{y_1, \ldots, y_n\}$ and $Z = \{z_1, \ldots, z_n\}$. A matching is a subset $M \subseteq \mathcal{C}$ such that no two elements in M agree in any coordinate, and the purpose of 3-DM is to answer the question: does there exist a perfect matching M on $\mathcal{C}$, that is, a matching of size n? This problem with the restriction that no element of $X \cup Y \cup Z$ occurs in more than 3 triples is known to be NP-complete (problem [SP1] page 221 in (Garey and Johnson 1979)).

Let $I = (\mathcal{C}, X \times Y \times Z)$ be an instance of 3-DM with $n \geq 3$. For $a \in X \cup Y \cup Z$, $d(a)$ denotes the number of occurrences of a in $\mathcal{C}$, that is the number of triples of $\mathcal{C}$ which contain a; we can assume that $\forall a \in X \cup Y \cup Z$, $d(a) \in \{2, 3\}$. From I, we build an instance of the voting problem as follows. The set C of candidates contains x^*, $C_1 = \{x'_i, y'_i, z'_i : 1 \leq i \leq n\} \cup \{c_e : e \in \mathcal{C}\}$ where x'_i, y'_i, z'_i correspond to elements of $X \cup Y \cup Z$, c_e correspond to triplets of $\mathcal{C}$ and a set C_2 of dummy candidates. The set V of voters contains $V_1 = \{v^e : e \in \mathcal{C}\}$ and a set V_2 of dummy voters. For each voter, we only indicate her four first candidates (in the order of preference). Thus, the vote of v^e is (c_e, x'_i, y'_j, z'_k) where $e = (x_i, y_j, z_k) \in \mathcal{C}$. The preference of dummy voters are such that (i) the score of the candidates in C verifies $\forall c \in X' \cup Y' \cup Z'$, $score_{\vec{s}}(c, P_C) = n+1$, $score_{\vec{s}}(x^*, P_C) = n$ and $\forall c \in C_2$, $score_{\vec{s}}(c, P_C) = 1$ (it is possible since $d(a) \in \{2, 3\}$) and (ii) any voter of V_2 contains at most one candidate of $\{x'_i, y'_i, z'_i : 1 \leq i \leq n\}$ in positions up to 4, and if he contains one, then it is certainly ranked in top position.

We claim that I admits a perfect matching $M \subseteq \mathcal{C}$ if and only if x^* becomes a possible winner by adding three new candidates y^*_i, $i = 1, 2, 3$.

Let y^*_i for $i = 1, 2, 3$ be the new candidates added ($y^*_i \notin C$). Since we cannot increase the score of x^*, we must decrease by one point the score of candidates of $X' \cup Y' \cup Z'$.

Let us focus on candidates in X'. In order to reduce the score of x'_i, we must modify the preferences of voters v^e (since, by (ii) we cannot decrease the score of x'_i using voters of V_2). By construction, each such voter must put y^*_1, y^*_2, y^*_3 in positions up to 4 and then, the score of y^*_i increases by 1 each time. Since there are n candidates in X', we deduce that $score_{\vec{s'}}(y^*_i, P) \geq n$ for every $i = 1, 2, 3$. Since, $score_{\vec{s'}}(x^*, P) \leq score_{\vec{s}}(x^*, P_C) = n$, we deduce that for each $i \in \{1, \ldots, n\}$, exactly one voter among those of v^e must put candidates y^*_1, y^*_2, y^*_3 in positions up to 4. Finally, since the score of candidates in $Y' \cup Z'$ must also decrease by 1, we deduce that x^* is a possible winner iff $M = \{e \in \mathcal{C} : y^*_1, y^*_2, y^*_3$ are in positions up to 4 for voter $v_e\}$ is a perfect matching of $\mathcal{C}$ (for the remaining voters, y^*_i is put in position at least 5 for every $i = 1, 2, 3$). ■

By Proposition 1, we know that the possible winner problem w.r.t. candidate addition for 1-approval (which coincides with plurality) is polynomial for any number of new candidates. By Proposition 2 we know that the K-approval case is also polynomial when a single new candidate shows up. By Proposition 3, we know the problem to be NP-complete for 4-approval and 3 new candidates. These three results leave some open questions: we do now know whether the problem is NP-hard for 2- and 3-approval, and we do not know whether it is NP-hard for k-approval with $k \geq 4$ and exactly two new candidates.

Borda

Let us now consider the Borda rule (r_B). Characterizing possible Borda winners when adding candidates is easy due to the following lemma:

Lemma 1 *Let P_X be a X-profile and let $y_1, \ldots, y_k$ be k new candidates. $x \in X$ is a possible winner for P_X wrt. the addition of k new candidates if and only if $r_B(P) = x$, where P is the profile on $X \cup \{y_1, \ldots, y_k\}$ obtained from P_X by putting $y_1, \ldots, y_k$ right below x (in an arbitrary order) in every vote of P_X.*

Proof: Let $Above(x_i)$ denote the set of candidates ranked above some candidate x_i in a vote. We show that it is never beneficial to put any of the new candidates (say y_i) anywhere but right below x in the new profile P, *i.e.* that this minimizes $score(z, P) - score(x, P)$ for all $z \in X \setminus \{x\}$. There are two cases to consider: (i) putting y_i candidate above x, and (ii) putting y_i at least two positions below x. Clearly (i) is not optimal, as $score(z, P) - score(x, P)$ increases for all $z \in Above(y_i)$ and remains constant for the other ones. As for (ii), $score(z, P) - score(x, P)$ reduces for $z \in X \setminus (Above(y_i) \cup \{x\})$ but remains constant for all $z \in Above(y_i)$. Thus, $|Above(y_i) \setminus Above(x)|$ should be minimized by putting y_i right below x. Finally, observe that we need not care about the scores of all the y_i, because x dominates any of them in P. ■

The following result then easily follows:

Proposition 4 *Let P_X be an n-voter profile on X, and $x \in X$. x is a possible winner for Borda with respect to* *the addition of k new candidates if and only if*

$$k \geq \max_{z \in X \setminus \{x\}} \frac{\max(0, score(z, P_X) - score(x, P_X))}{N_{P_X}(x, z)}$$

Proof: Consider the case of one new candidate y. By Lemma 1 we know that y has to be placed right below x in the $X \cup \{y\}$−profile, hence $score(x, P_{X\cup\{y\}}) = score(x, P_X)+1$, and similarly for the candidates above x. For all candidates ranked below x on the other hand, we have $score(x, P_{X\cup\{y\}}) = score(x, P_X)$. Checking whether x is a possible winner then amounts to check, for each other candidate z, whether there are enough votes where x is prefered to z to compensate the score difference with this candidate, *i.e.* $N_{P_X}(x, z) \geq max(0, score(z, P_X) - score(x, P_X)$. This immediately generalizes to k new candidates. ∎

This means that possible winners with respect to adding any number of new candidates can be computed in polynomial time in the case of the Borda rule. Note that the general problem of computing possible winners for Borda is NP-complete (Xia and Conitzer 2008), therefore, the restriction of the problem to candidate addition induces a complexity fall.

We give an example to illustrate this result. Suppose we have $X = \{a, b, c, d\}$, $n = 4$, and an initial profile $P = \langle bacd, bacd, bacd, dacb\rangle$.
Let us denote by $\delta(x, z)$ the expression

$$\frac{\max(0, score(z, P_X) - score(x, P_X))}{N_{P_X}(x, z)}$$

Hence the values of $\delta(x, z)$ for any pair $x, z \in X$:
$\delta(b, x) = 0$ for any $x \neq b$;
$\delta(a, b) = \frac{9-8}{1} = 1$; $\delta(a, c) = \delta(a, d) = 0$;
$\delta(c, a) = \frac{8-4}{0} = +\infty$; $\delta(c, b) = \frac{9-4}{1} = 5$; $\delta(c, d) = 0$;
$\delta(d, a) = \frac{8-3}{1} = 5$; $\delta(d, b) = \frac{9-3}{1} = 6$; $\delta(d, c) = \frac{4-3}{1} = 1$.
Therefore:

- with at least 1 new candidate, a is a possible winner;
- with at least 6 new candidates, d also becomes a possible winner;
- whatever the number of new candidates, c is never a possible winner.

Notice that c is never a possible winner although it has a higher Borda score than d in P_X.

Lemma 4 and Proposition 4 still hold under the following more general conditions: for all $i \in \{0..p-2\}$: $(s_i - s_{i+1}) \leq (s_{i+1} - s_{i+2})$. In words, this corresponds to rules where the difference of scores between successive ranks can only become smaller or remain constant as we come closer to the highest ranks. This condition is satisfied by Borda (but not by plurality), by veto, and by rules such as "lexicographic veto", where the scoring vector is $\langle M^p, M^p - M, M^p - M^2, \ldots, M^p - M^{p-1}, 0\rangle$ where $M > n$. For such rules, the possible winner problem remains polynomial, whatever the number of new candidates.

Hardness with a single new candidate

Even though we have seen that the possible winner problem can be NP-hard for some scoring rules, NP-hardness required the addition of several new candidates. We now show that there exists a scoring rule for which the possible winner problem is NP-hard with respect to the addition of *one* new candidate.

The scoring rule we use is very simple: it allows each voter to approve exactly 3 candidates, and offers 3 different levels of approval (assigning respectively 3,2,1 points to the three preferred candidates). Let r_Δ be the scoring rule defined by the vector $\langle 3, 2, 1, 0, \ldots, 0\rangle$ with $p - 3$ 0's completing the vector.

Proposition 5 *Deciding if x^* is a possible winner for r_Δ with respect the addition of one candidate is* NP*-complete.*

Proof: This problem is clearly in NP. The proof is quite similar to that of Proposition 3. Let $I = (\mathcal{C}, X \times Y \times Z)$ be an instance of 3-DM with $n \geq 5$ and $\forall a \in X \cup Y \cup Z$, $d(a) \in \{2, 3\}$. From I, we build an instance of the voting problem as follows. The set C of candidates contains x^*, $C_1 = \{x'_i, y'_i, z'_i : 1 \leq i \leq n\}$ where x'_i, y'_i, z'_i correspond to elements of $X \cup Y \cup Z$ and a set C_2 of dummy candidates. The set V of voters contains $V_1 = \{v^e : e \in \mathcal{C}\}$ and a set V_2 of dummy voters. For each voter $i \in V$, we only indicate the vote for the 3 first candidates. So, the vote (a_1, a_2, a_3) means that candidate a_i receives $4-i$ points. Thus, the vote of v^e is (x'_i, y'_j, z'_k) where $e = (x_i, y_j, z_k) \in \mathcal{C}$. The preferences of dummy voters are such that (i) the score of the candidates in C verifies $\forall c \in C_1$, $score_{\vec{s}}(c, P_C) = 3n + 1$, $score_{\vec{s}}(x^*, P_C) = 3n$ and $\forall c \in C_2$, $score_{\vec{s}}(c, P_C) \leq 3$ and (ii) any voter of V_2 contains at most one candidate of $\{x'_i, y'_i, z'_i : 1 \leq i \leq n\}$ in positions up to 3, and if he contains one in second position, then x^* occurs in third position.

We claim that I admits a perfect matching $M \subseteq \mathcal{C}$ if and only if x^* becomes a possible winner by adding a new candidate y^*.

Following a line of reasoning similar to the one developed in the proof of Proposition 3, we conclude that for each $i \in \{1, \ldots, n\}$, exactly one voter among those of v^e must put candidate y^* in top position. Since the score of $Y' \cup Z'$ must also decrease by 1, we deduce that x^* is a possible winner iff $M = \{e \in \mathcal{C} : y^*$ is in top position for voter $v_e\}$ is a perfect matching of $\mathcal{C}$ (for the remaining voters, y^* is put in last position). ∎

This rule is an example for which it is difficult to identify possible winners with a single missing candidate. Giving a characterization of those rules sharing this property is an open problem. Note also that this does not necessarily imply that the problem is difficult for any number of new candidates. Indeed, it is worthy to observe that the possible winners problem may become simple again when sufficiently many new candidates are added (this is true in particular when that number is unbounded). We leave this aspect of the problem for future work.

Conclusion

In this paper we have considered voting situations where new candidates may show up during the process. This problem increasingly occurs in our societies, as many votes now take place online (through dedicated platforms, or simply by email exchange) during an extended period of time.

We have identified the computational complexity of computing possible winners for some scoring rules. Some of them allow polynomial algorithms for the problem (*e.g.* plurality, Borda, veto). For some others, the number of new candidates is an important parameter: K-approval is polynomial when one single candidate is added, but 4-approval is NP-complete as soon as 3 new candidates show up. Cases in between are currently under investigation. Finally, we have exhibited a simple rule where the problem is hard for only one new candidate.

The results make the assumption of the most favourable tie-breaking. In the other extreme case (if we want x to be a strict winner, *i.e.* win regardless of the tie-breaking rule), the results are easily adapted: the inequalities in Prop. 1 and 4 become strict. For K-approval, the first condition of Prop. 2 becomes strict but the second one should now read $S_K(P_X, x) \geq \sum_{x_i \in X} \max(0, S_K(P_X, x_i) - S_K(P_X, x) + 1)$. As for veto, all other initial candidates need to be vetoed at least once. A more general treatment would require cumbersome expressions, and is also somewhat problematic since the identity of the new candidate is not known anyway (making it difficult to specify easily a tie-breaking rule on these candidates).

As for future work, a first direction to follow would consist in trying to obtain more general results for scoring rules, as do (Betzler and Dorn 2009) for the general version of the possible winner problem. Extending the study to other families of voting rules, such as rules based on the majority graph, is also worth investigating.

Of course, identifying possible winners is not the end of the story. In practice, one may for instance also be interested in a refinement of this notion: knowing how likely it is that a given candidate will win. Another interesting issue consists in designing elicitation protocols when the preferences about the 'old' candidates are already known. In this case, a trade-off occurs between the storage cost and communication cost, since keeping track of more information is likely to help to reduce the burden of elicitation.

Acknowledgments. We would like to thank three anonymous reviewers for their very useful comments, as well as the audience of the "Computational Foundations of Social Choice" Dagstuhl seminar (10101) for informative feedback which helped to improve this paper. This work has been partially supported by the ANR-funded project "Computational Social Choice" (ComSoc).

References

Bartholdi, J.; Tovey, C.; and Trick, M. 1992. How hard is it to control an election? *Social Choice and Welfare* 16(8-9):27–40.

Betzler, N., and Dorn, B. 2009. Towards a dichotomy of finding possible winners in elections based on scoring rules. In *Proc. MFCS 2009*, volume 5734 of *Lecture Notes in Computer Science*, 124–136. Springer.

Betzler, N.; Hemmann, S.; and Niedermeier, R. 2009. A multivariate complexity analysis of determining possible winners given incomplete votes. In *Proc. IJCAI-09*, 53–58.

Chevaleyre, Y.; Lang, J.; Maudet, N.; and Ravlly-Abadie, G. 2009. Compiling the votes of a subelectorate. In *Proceedings of IJCAI-09*, 97–102.

Conitzer, V., and Sandholm, T. 2002a. Complexity of manipulating elections with few candidates. In *Proceedings of AAAI-02*, 314–319.

Conitzer, V., and Sandholm, T. 2002b. Vote elicitation: complexity and strategy-proofness. In *Proceedings of AAAI-02*, 392–397.

Elkind, E.; Faliszewski, P.; and Slinko, A. 2010. Cloning in elections. In *Proceedings of AAAI-10*. To appear.

Faliszewski, P.; Hemaspaandra, E.; and Hemaspaandra, L. 2009. Multimode control attacks on elections. In *Proceedings of IJCAI-09*, 128–133.

Garey, M., and Johnson, D. 1979. *Computers and intractability. A guide to the theory of NP-completeness*. Freeman.

Hazon, N.; Aumann, Y.; Kraus, S.; and Wooldridge, M. 2009. Evaluation of election outcomes under uncertainty. In *Proceedings of AAMAS-08*, 959–966.

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

Konczak, K., and Lang, J. 2005. Voting procedures with incomplete preferences. In *Proc. IJCAI-05 Multidisciplinary Workshop on Advances in Preference Handling*.

Pini, M.; Rossi, F.; Venable, K. B.; and Walsh, T. 2007. Incompleteness and incomparability in preference aggregation. In *Proceedings of IJCAI'07*, 1464–1469.

Walsh, T. 2008. Complexity of terminating preference elicitation. In *Proceedings of AAMAS-08*, 967–974.

Xia, L., and Conitzer, V. 2008. Determining possible and necessary winners under common voting rules given partial orders. In *Proceedings of AAAI-08*, 196–201.

Xia, L.; Zuckerman, M.; Procaccia, A.; Conitzer, V.; and Rosenschein, J. 2009. Complexity of unweighted coalitional manipulation under some common voting rules. In *Proceedings of IJCAI-09*, 348–353.