

# A Broader Picture of the Complexity of Strategic Behavior in Multi-Winner Elections

Reshef Meir, Ariel D. Procaccia, Jeffrey S. Rosenschein

School of Engineering and Computer Science  
The Hebrew University of Jerusalem  
Jerusalem, Israel  
{reshef24, arielpro, jeff}@cs.huji.ac.il

## ABSTRACT

Recent work by Procaccia, Rosenschein and Zohar [14] established some results regarding the complexity of manipulation and control in elections with multiple winners, such as elections of an assembly or committee; that work provided an initial understanding of the topic. In this paper, we paint a more complete picture of the topic, investigating four prominent multi-winner voting rules. First, we characterize the complexity of manipulation and control in these voting rules under various kinds of formalizations of the manipulator's goal. Second, we extend the results about complexity of control to various well-known types of control. This work enhances our comprehension of which multi-winner voting rules should be employed in various settings.

## Categories and Subject Descriptors

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

## General Terms

Algorithms, Theory, Economics

## Keywords

Computational complexity, Voting

## 1. INTRODUCTION

Game theory is chiefly concerned with the reaction of rational entities to incentives. These entities can, naturally, be people (who, in practice, may not act rationally), but, alternatively, can be computational agents, driven by pristine calculations of utility. The theory of social choice, in particular, has long struggled with the following problem: is it possible to prevent strategic behavior on the part of the participants in an election (the voters, or the authority conducting the election)?

**Cite as:** A Broader Picture of the Complexity of Strategic Behavior in Multi-Winner Elections, R. Meir, A. D. Procaccia and J. S. Rosenschein, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., 2008, Estoril, Portugal, pp.991-998.  
Copyright © 2008, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

In an election the voters are asked to report their preferences over candidates. A voter is said to *manipulate* the election when he reports false preferences, in an attempt to influence the outcome of the election. The election's result is determined by a voting rule, which designates a winning candidate given the voters' preferences. The Gibbard-Satterthwaite theorem [9] asserts that any voting rule which cannot be manipulated must be a dictatorship, i.e., there is one voter who dictates the outcome of the election.<sup>1</sup> A considerable body of work has been devoted to circumventing this theorem. Some of the approaches considered in economics are restrictions of the agents' preferences, or assuming that money is available, which leads to mechanism design solutions.

An equally sinister setting is the one where the authority controlling the election, referred to as the *chairman*, attempts to *control* the outcome of the election by tampering with the list of registered voters or the slate of candidates. For instance, the chairman might register additional voters who support his favorite candidate, or demolish the competition by disqualifying some of the candidates.

## 1.1 A Computational Approach

In a series of important papers, Bartholdi, Tovey and Trick [2, 3] have argued that *computational complexity* can be a barrier against strategic behavior in elections. Indeed, although manipulation and control may be possible in theory, in practice they can amount to solving  $\mathcal{NP}$ -hard problems, suggesting that the voters or the chairman might as well avoid cheating altogether.

Indeed, in the context of manipulation, the well-known Single Transferable Vote (STV) rule has long been known to be  $\mathcal{NP}$ -hard to manipulate [1]. More recent papers show that common voting rules can be enhanced in a way that makes them hard to manipulate [4, 7], or that prominent rules are hard to manipulate in alternative settings, such as manipulation by coalitions of voters (see, e.g., [6]).

The computational aspects of control have also received attention, for the same reasons mentioned above. Inspired by Bartholdi, Tovey and Trick [3], very recent papers have extended their results to various types of control and a variety of voting rules [11, 10, 8].

<sup>1</sup>This is a simplification, as the theorem requires some assumptions. Most importantly, there must be at least 3 candidates, and the domain of the agents' preferences is the domain of all possible linear orders over the candidates.

## 1.2 Multi-winner elections and our results

The purpose of multi-winner elections is to choose a committee or assembly (for instance, elections for parliament). A multi-winner voting rule maps the preferences of the voters to subsets of candidates, and effectively specifies the composition of the assembly. We note that this structured setting, where voters essentially have preferences over subsets of candidates, is related to recent work on combinatorial voting (see, e.g., Lang [12] and the references listed there).

Now, the properties that are considered especially desirable with respect to multi-winner voting rules are not necessarily the ones usually sought in single-winner voting rules, and therefore different rules are usually investigated. The four prominent rules we shall examine here are Single Non-Transferable Vote (SNTV), Bloc voting, Approval, and Cumulative voting (for more details, see Section 2).

Some results have recently been established by Procaccia, Rosenschein and Zohar [14] regarding the complexity of manipulation and control in multi-winner elections. These results can be summarized as follows:

In...	MANIPULATION	CONTROL (Add Voters)
SNTV	$\mathcal{P}$	$\mathcal{P}$
Bloc	$\mathcal{P}$	$\mathcal{NP}\text{-c}$
Approval	$\mathcal{P}$	$\mathcal{NP}\text{-c}$
Cumulative	$\mathcal{NP}\text{-c}$	$\mathcal{NP}\text{-c}$

All the results hold for a very general formulation of the computational problems. The manipulator/chairman assigns utilities to different candidates; can he manipulate/control the election in a way that the total (additive) utility of the set of winners is above a given threshold? Our first goal is to extend all these results to more restricted questions. Is it possible to include a favorite candidate in the set of winners? Perhaps it is possible to determine completely the set of winners? Or maybe it is possible to just include as many favorite candidates as possible among the winners?

In addition, Procaccia et al. have only looked at one specific type of control, control by adding voters. We extend the results to three other types of control (introduced in Bartholdi et al. [3]), namely control by removing voters, by adding candidates, and by removing candidates. Impatient readers can jump directly to a summary of our results, available below as Tables 1 and 2.

## 2. PRELIMINARIES

We now introduce some notation, as well as the four multi-winner voting rules that we shall investigate.

Let the set of voters be  $V = \{v_1, v_2, \dots, v_n\}$ ; let the set of candidates be  $C = \{c_1, c_2, \dots, c_m\}$ . We denote the number of seats in the assembly—the number of candidates to be elected—by  $k \in \mathbb{N}$ .

Multi-winner voting rules differ from single-winner rules in the properties that they are expected to satisfy. A major concern in multi-winner elections is *proportional representation*: a faction that consists of a fraction  $X$  of the population should be represented by approximately a fraction  $X$  of the seats in the assembly. This property is not satisfied by (generalizations of) many of the rules usually considered with respect to single-winner elections.

So, here we examine four of the prevalent multi-winner voting rules. In all four, the candidates are awarded points

by the voters, and the candidates with the most points win the election.

- *Single Non-Transferable Vote (SNTV)*: each voter gives one point to a favorite candidate.
- *Bloc voting*: each voter gives one point to each of  $k$  candidates.
- *Approval voting*: each voter can approve or disapprove any candidate; an approved candidate is awarded one point, and there is no limit to the number of candidates a voter can approve.
- *Cumulative voting*: allows voters to express intensities of preferences, by asking them to distribute a fixed number of points among the candidates. We denote the fixed pool of points by  $L$ .

*Scoring rules* are a prominent family of voting rules. A voting rule in this family is defined by a real vector  $\vec{\alpha} = \langle \alpha_1, \dots, \alpha_m \rangle$ , where  $\alpha_l \geq \alpha_{l+1}$  for  $l = 1, \dots, m - 1$ . Each voter reports a ranking of the candidates, thus awarding  $\alpha_1$  points to the top-ranked candidate,  $\alpha_2$  points to the second candidate, and in general  $\alpha_l$  points to the candidate ranked in place  $l$ . Note that SNTV is the scoring rule defined by the vector  $\langle 1, 0, \dots, 0 \rangle$ , and Bloc is the scoring rule defined by vector  $\langle 1, \dots, 1, 0, \dots, 0 \rangle$ , where the number of 1's is  $k$ .

## 3. MANIPULATION

Manipulation in voting is considered to be any scenario in which a voter reveals false preferences in order to improve the outcome of the election. This has various negative consequences; not only do voters spend valuable computational resources determining which lie to employ, but worse, the outcome may not be one that reflects the social good. Presumably, a voting rule which is hard-to-manipulate *a priori* precludes such undesirable behavior. A general definition of the manipulation problem in multi-winner elections was given in Procaccia et al. [14]:

**DEFINITION 3.1.** In the ( $k$ -winner) MANIPULATION problem, we are given a set  $C$  of candidates, a set  $V$  of voters and their ballots, the number of winners  $k \in \mathbb{N}$ , a utility function  $u : C \rightarrow \mathbb{Z}$ , and an integer  $t \in \mathbb{Z}$ . We are asked whether a single additional voter (the manipulator) can cast his vote in a way that in the resulting election,  $\sum_{c \in W} u(c) \geq t$ , where  $W$  is the set of winners of size  $k$ .

**REMARK 3.2.** The Manipulator's utility function is implicitly assumed to be additive. One can consider more elaborate utility functions, such as the ones investigated in the context of combinatorial auctions, but that is beyond the scope of this paper.

Procaccia et al. [14] have established that this problem is tractable in SNTV, Bloc, and Approval, but that it is  $\mathcal{NP}$ -complete in Cumulative voting. Although Cumulative voting has emerged as the winner in this complexity-theoretic competition, one might argue that the general formulation of the problem given above makes manipulation harder. Indeed, the manipulator might have the following, more specific, goals in mind.

1. The manipulator has a specific candidate whom he is interested in seeing among the winners.

2. The manipulator has a favorite subset of candidates, and he is interested in seeing *all of them* among the winners.
3. The manipulator has a favorite subset of candidates, and he is interested in seeing *as many as possible of them* among the winners.

Notice that the third setting is a special case of Definition 3.1—indeed, simply restrict  $u$  to be a boolean-valued function, i.e.,  $u : C \rightarrow \{0, 1\}$ . Furthermore, the second setting is a special case of the third since, if  $D \subseteq C$  is the favorite subset, we can set

$$u(c) = 1 \Leftrightarrow c \in D$$

and  $t = |D|$ . The first is a special case of the second when  $|D| = 1$ .

REMARK 3.3. In all manipulation and control problems, we assume tie-breaking is *adversarial* to the manipulator or chairman, i.e., ties are broken in favor of candidates with lower utility. This is a standard assumption, made in many of the papers on these topics [6, 14].

The next proposition gives a negative answer to the question of whether MANIPULATION in Cumulative voting is still hard in the abovementioned settings. Indeed, we put forward an algorithm that decides the problem under any boolean-valued utility function.

PROPOSITION 3.4. MANIPULATION in Cumulative voting with any boolean-valued utility function  $u : C \rightarrow \{0, 1\}$  is in  $\mathcal{P}$ .

PROOF. Let  $s[c]$  be the score of candidate  $c \in C$  before the manipulator has cast his vote, and  $s^*[c]$  be  $c$ 's score when the manipulator's vote is taken into account. Assume without loss of generality that  $s[c_1] \geq s[c_2] \geq \dots \geq s[c_m]$ . Let  $D = \{d_1, d_2, \dots\}$  be the set of desirable candidates  $d \in C$  with  $u(d) = 1$ , and again assume these are sorted by nonincreasing scores.

Informally, we are going to find a threshold  $thresh$  such that pushing  $t$  candidates above the threshold guarantees their victory. Then we will check whether it is possible to distribute  $L$  points such that at least  $t$  candidates pass this threshold, where  $L$  is the number of points available to each voter.

Formally, consider Algorithm 1 (w.l.o.g.  $k \geq t$ , otherwise manipulation is impossible). The algorithm clearly halts in polynomial time. It only remains to prove the correctness of the algorithm.

LEMMA 3.5. *The above algorithm correctly decides MANIPULATION in Cumulative voting with any boolean-valued utility function.*

PROOF. Denote by  $\hat{W} = \{c_1, \dots, c_k\}$  the  $k$  candidates with highest score (sorted) before the manipulator's vote, and by  $W$  the final set of  $k$  winners. The threshold candidate  $c_{j^*}$  partitions  $\hat{W}$  into two disjoint subsets:  $\hat{W}_u = \{c_1, \dots, c_{j^*-1}\}$ ,  $\hat{W}_d = \{c_{j^*}, \dots, c_k\}$ . By the maximality of  $j^*$ , it holds that:

$$|\hat{W}_u \cap D| + |\hat{W}_d| = |\hat{W}_u \cap D| + (k + 1 - j^*) = t. \quad (1)$$

Note that  $S$  is the exact number of votes required to push  $t$  desirable candidates above the threshold. Now, we must

---

**Algorithm 1** Decides MANIPULATION in Cumulative voting with boolean valued utility

---

```

1:  $j^* \leftarrow \max\{j : |\{c_1, c_2, \dots, c_{j-1}\} \cap D| + k + 1 - j \geq t \text{ and } c_j \notin D\}$ 
    $\triangleright j^*$  exists, since the condition holds for the first candidate not in  $D$ 
2:  $thresh \leftarrow s[c_{j^*}]$ 
3:  $S \leftarrow \sum_{j=1}^t \max\{0, thresh + 1 - s[d_j]\}$ 
4: if  $S \leq L$  then
5:   return true
6: else
7:   return false
8: end if

```

---

show that the manipulator can cast his vote in a way that the winner set  $W$  satisfies  $|W \cap D| \geq t$  if, and only if,  $L \geq S$ .

Suppose first that  $S \leq L$ . Then it is clearly possible to push  $t$  desirable candidates above  $thresh$ .  $\hat{W}_u \cap D$  were above the threshold already; it follows that  $\hat{W}_d$  was replaced entirely by desirable candidates.

Let  $W = \{w_1, \dots, w_k\}$  be the set of *new* winners. In particular, we can write  $W = \hat{W}_u \uplus \{w_{j^*}, \dots, w_k\}$ .  $\hat{W}_u$  contains  $|\hat{W}_u \cap D|$  desirable candidates, while  $\{w_{j^*}, \dots, w_k\}$  consists purely of desirable candidates. By Equation (1):

$$\begin{aligned} |W \cap D| &= |\hat{W}_u \cap D| + |\{w_{j^*}, \dots, w_k\}| \\ &= |\hat{W}_u \cap D| + |\hat{W}_d| \\ &= t \end{aligned}$$

Conversely, suppose  $S > L$ . We must show that the manipulator cannot distribute  $L$  points in a way that  $t$  candidates from  $D$  are among the winners.

Clearly there is no possibility to push  $t$  desirable candidates above  $thresh$ . Consider some ballot cast by the manipulator, and assume w.l.o.g. that the manipulator distributed points only among the candidates in  $D$ . Denote the new set of winners by  $W = W_u \uplus W_d$ , where

$$W_u = \{c \in C : s^*[c] > thresh\}$$

$$W_d = \{c \in C : s^*[c] \leq thresh\}.$$

We claim that

$$|W_u \cap \bar{D}| = k - t, \quad (2)$$

where  $\bar{D} = C \setminus D$ . Indeed, by Equation (1)

$$|\hat{W}_u \cap D| = t - k - 1 + j^*.$$

Since no votes were awarded to candidates in  $\bar{D}$ ,

$$\begin{aligned} |W_u \cap \bar{D}| &= |\hat{W}_u \cap \bar{D}| = |\hat{W}_u| - |\hat{W}_u \cap D| \\ &= (j^* - 1) - (t - k - 1 + j^*) \\ &= k - t \end{aligned}$$

Denote by  $F$  the set of candidates that were pushed above the threshold. Formally:

$$F = \{c \in D : s[c] \leq thresh \text{ and } s^*[c] > thresh\}$$

Thus:

$$W_u = \hat{W}_u \uplus F.$$

Let  $w^*$  be the new position of candidate  $c_{j^*}$  when the candidates are sorted by nonincreasing  $s^*[c]$ . It holds that

$$w^* = j^* + |F|.$$

We now claim that

$$|W_d \cap \bar{D}| \geq 1. \quad (3)$$

Recall that there are less than  $t$  desirable candidates above the threshold, thus:

$$\begin{aligned} |W_u \cap D| &< t && \Rightarrow \\ |\hat{W}_u \cap D| + |F| = |W_u \cap D| &< t = |\hat{W}_u \cap D| + k + 1 - j^* && \Rightarrow \\ |F| &< k + 1 - j^* && \Rightarrow \\ w^* = j^* + |F| &< j^* + k + 1 - j^* = k + 1 && \Rightarrow \\ w^* &\leq k && \Rightarrow \\ c_{j^*} &\in W_d && \Rightarrow \\ |W_d \cap \bar{D}| &\geq 1 \end{aligned}$$

By combining Equations (2) and (3), we finally obtain:

$$\begin{aligned} |W \cap D| &= k - |W \cap \bar{D}| \\ &= k - (|W_u \cap \bar{D}| + |W_d \cap \bar{D}|) \\ &\leq k - (k - t + 1) \\ &= t - 1 \\ &< t \end{aligned}$$

□

The proof of Proposition 3.4 is completed. □

REMARK 3.6. The proof shows that the manipulation of Cumulative voting by a *coalition* of (even weighted) voters, as in Conitzer et al. [6], is tractable under a boolean-valued utility function. This follows by simply joining the (weighted) score pools of all the voters in the coalition.

SNTV and Bloc voting, which are both scoring rules, are known to be easy to manipulate under a general utility function [14]. The next proposition establishes that this is true for any scoring rule, under a boolean-valued utility function.

PROPOSITION 3.7. *Let  $P$  be a scoring rule defined by the parameters  $\bar{\alpha} = \langle \alpha_1, \dots, \alpha_m \rangle$ . MANIPULATION in  $P$  with any boolean-valued utility function  $u : C \rightarrow \{0, 1\}$  is in  $P$ .*

PROOF. Let  $\bar{\alpha} = \langle \alpha_1, \dots, \alpha_m \rangle$  be the parameters of the scoring rule in question. Denote the score of each candidate  $c \in C$ , before the manipulator has cast his vote, by  $s[c]$ . Let  $\mathcal{J}$  be the manipulator's preference profile, given by:

$$\mathcal{J} = c_{j_1} \succ c_{j_2} \succ \dots \succ c_{j_m}$$

Suppose some candidate  $c \in C$  was ranked in place  $l$  by the manipulator,  $c = c_{j_l}$ . Denote the final score of candidate  $c$ , according to the manipulator's profile  $\mathcal{J}$ , by:

$$s_{\mathcal{J}}[c] = s[c] + \alpha_l$$

Finally, denote the winner set that results from the manipulator's ballot  $\mathcal{J}$  by  $W_{\mathcal{J}}$ .

LEMMA 3.8. *Given  $C' \subseteq C$ ,  $|C'| = k$ , it is possible to determine in polynomial time if there exists  $\mathcal{J}$  s.t.  $C' = W_{\mathcal{J}}$ .*

PROOF. Denote  $C' = \{c'_1 \dots c'_k\}$ ,

$$C'' = C \setminus C' = \{c''_1, \dots, c''_{m-k}\},$$

where both  $C', C''$  are sorted by nondecreasing score  $s[c]$ . Let

$$\mathcal{J}^* = c'_1 \succ c'_2 \succ \dots \succ c'_k \succ c''_1 \succ \dots \succ c''_{m-k}$$

This preference profile ranks the players in  $C'$  first, while giving more points to candidates with lower initial score. Candidates from  $C''$  are ranked next, and the same rule applies. The intuition is that we would like the candidates in  $C'$  to have a high-as-possible, more or less balanced, score. Likewise, we would like the candidates in  $C''$  to have a low-as-possible balanced score. This strategy generalizes the algorithm of Bartholdi et al. [2].

We claim that there exists  $\mathcal{J}$  s.t.  $C' = W_{\mathcal{J}}$  iff  $C' = W_{\mathcal{J}^*}$ . If  $C' = W_{\mathcal{J}^*}$  then obviously there exists  $\mathcal{J}$  s.t.  $C' = W_{\mathcal{J}}$ . Conversely, suppose there exists some  $\mathcal{J}^{\#}$  such that  $C' = W_{\mathcal{J}^{\#}}$ . Without loss of generality, this holds (by the adversarial tie breaking assumption)<sup>2</sup> iff

$$\forall c' \in C', c'' \in C'', s_{\mathcal{J}^{\#}}[c''] < s_{\mathcal{J}^{\#}}[c']. \quad (4)$$

We argue that it is possible to obtain  $\mathcal{J}^*$  from  $\mathcal{J}^{\#}$  by iteratively transposing pairs of candidates, without changing the winner set. Indeed, we distinguish between three cases:

1.  $\exists j_1, j_2 \in \{1, 2, \dots, k\}$  such that  $s[c'_{j_1}] > s[c'_{j_2}]$ , but in  $\mathcal{J}^{\#}$  it holds that  $c'_{j_1} \succ c'_{j_2}$ . Now, transpose the rankings of  $c'_{j_1}$  and  $c'_{j_2}$  in  $\mathcal{J}^{\#}$ , i.e., consider the preference profile which is identical to  $\mathcal{J}^{\#}$  except that the places of  $c'_{j_1}$  and  $c'_{j_2}$  are switched. Denote by  $W$  the new set of winners.

The score of  $c'_{j_2}$  increased, so he is certainly still in  $W$ . Moreover, the new final (possibly lower) score of  $c'_{j_1}$  is:

$$s[c'_{j_1}] + \alpha_{j_2} \geq s[c'_{j_2}] + \alpha_{j_2} = s_{\mathcal{J}^{\#}}[c'_{j_2}]$$

By (4) we have that:

$$\forall c'' \in C'', s_{\mathcal{J}^{\#}}[c''] < s_{\mathcal{J}^{\#}}[c'_{j_2}]$$

Therefore,  $c'_{j_1} \in W$  even after the transposition. We conclude that it still holds that  $C' = W$ .

2.  $\exists j_1, j_2 \in \{1, 2, \dots, m - k\}$  such that  $s[c'_{j_1}] > s[c'_{j_2}]$ , but in  $\mathcal{J}^{\#}$  it holds that  $c'_{j_1} \succ c'_{j_2}$ . A similar argument holds in this case.
3.  $\exists c' \in C', c'' \in C''$  such that in  $\mathcal{J}^{\#}$  it holds that  $c'' \succ c'$ . Clearly the desirable candidate  $c'$  can only rank higher if we transpose the two candidates.

Using the three types of transpositions, we can replace a couple of candidates at each step until we obtain  $\mathcal{J}^*$  from  $\mathcal{J}^{\#}$ . In each such step it remains true that  $C' = W$ , thus  $C' = W_{\mathcal{J}^*}$ . □

LEMMA 3.9. *Given  $C' \subseteq C$ ,  $|C'| \leq k$ , it is possible to determine in polynomial time if there exists  $\mathcal{J}$  s.t.  $C' \subseteq W_{\mathcal{J}}$ .*

PROOF. Let  $C' \subseteq C$ ,  $|C'| = k' < k$ . We add to  $C'$  the  $k - k'$  candidates from  $C''$  with the highest score (according to  $s[c]$ ), and denote this new set of size  $k$  by  $C^*$ . According to Lemma 3.8, we can determine efficiently if there exists  $\mathcal{J}$  such that  $C^* = W_{\mathcal{J}}$ .

We argue that it is enough to check  $C^*$ . Indeed, assume that there exists  $\mathcal{J}$  such that  $C' \subseteq W_{\mathcal{J}}$ . Let  $c \in C \setminus W_{\mathcal{J}}$  such that there exists  $c' \in W_{\mathcal{J}}$  with  $s[c'] < s[c]$ . Now, if we

<sup>2</sup>Tie breaking works against candidates with utility 1 (which are the ones we ultimately care about), but in favor of candidates in  $C'$  with utility 0. However, for ease of exposition, we do not deal with such borderline cases here.







