Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)

Complexity of Manipulation with Partial Information in Voting

Palash Dey⋆, Neeldhara Misra†, Y. Narahari⋆
⋆Indian Institute of Science, Bangalore
†Indian Institute of Technology, Gandhinagar
⋆{palash|hari}@csa.iisc.ernet.in, †mail@neeldhara.com

Abstract

The Coalitional Manipulation problem has been studied extensively in the literature for many voting rules. However, most studies have focused on the complete information setting, wherein the manipulators know the votes of the non-manipulators. While this assumption is reasonable for purposes of showing intractability, it is unrealistic for algorithmic considerations. In most real-world scenarios, it is impractical for the manipulators to have accurate knowledge of all the other votes. In this paper, we investigate manipulation with incomplete information. In our framework, the manipulators know a partial order for each voter that is consistent with the true preference of that voter. In this setting, we formulate three natural computational notions of manipulation, namely weak, opportunistic, and strong manipulation. We consider several scenarios for which the traditional manipulation problems are easy (for instance, Borda with a single manipulator). For many of them, the corresponding manipulative questions that we propose turn out to be computationally intractable. Our hardness results often hold even when very little information is missing, or in other words, even when the instances are quite close to the complete information setting. Our overall conclusion is that computational hardness continues to be a valid obstruction to manipulation, in the context of a more realistic model.

1 Introduction

In many real life and AI related applications, agents often need to agree upon a common decision although they have different preferences over the available alternatives. A natural tool used in these situations is voting. In a typical voting scenario, we have a set of candidates and a set of voters reporting their rankings of the candidates called their preferences or votes. A voting rule selects one candidate as the winner once all voters provide their votes. A central issue in voting is the possibility of *manipulation*. For many voting rules, it turns out that even a single vote, if cast differently, can alter the outcome. In particular, a voter manipulates the voting rule if, by misrepresenting her preference, she obtains an outcome that she prefers over the "honest" outcome. In a cornerstone impossibility result, [Gibbard, 1973; Satterthwaite, 1975] show that every unanimous and non-dictatorial voting rule with three candidates or more is manipulable. We refer to [Brandt *et al.*, 2015] for an excellent introduction to issues in computational social choice theory.

1.1 Background

Considering that voting rules are indeed susceptible to manipulation, it is natural to seek ways by which elections can be protected from manipulations. The works of Bartholdi et al. [Bartholdi *et al.*, 1989; Bartholdi and Orlin, 1991] approach the problem from the perspective of computational intractability. They exploit the possibility that voting rules, despite being vulnerable to manipulation in theory, may be hard to manipulate in practice. Indeed, a manipulator is faced with the following decision problem: given a collection of votes $\mathcal{P}$ and a distinguished candidate c, does there exist a vote v that, when tallied with $\mathcal{P}$, makes c win for a (fixed) voting rule r? The manipulation problem has subsequently been generalized to the problem of COALITIONAL MANIPULATION (CM) by Conitzer et al. [Conitzer *et al.*, 2007], where one or more manipulators collude together and try to make a distinguished candidate win. The manipulation problem, fortunately, turns out to be NP-hard in several settings. This established the success of the approach of demonstrating a computational barrier to manipulation.

However, despite having set out to demonstrate the hardness of manipulation, the initial results in [Bartholdi *et al.*, 1989] were to the contrary, indicating that many voting rules are in fact easy to manipulate. Moreover, even with multiple manipulators involved, popular voting rules like plurality, veto, k-approval, Bucklin, and Fallback continue to be easy to manipulate [Xia *et al.*, 2009]. While we know that the computational intractability may not provide a strong barrier [Procaccia and Rosenschein, 2007b; 2007a; Walsh, 2011; Isaksson *et al.*, 2012; Dey, 2015; Dey *et al.*, 2016; 2015a; Dey and Narahari, 2015, and references therein] even for rules for which the coalitional manipulation problem turns out to be NP-hard, in all other cases the possibility of manipulation is a much more serious concern.

1.2 Motivation and Problem Formulation

In our work, we propose to extend the argument of computational intractability to address the cases where the approach appears to fail. We note that most incarnations of the manipulation problem studied so far are in the complete information setting, where the manipulators have complete knowledge of the preferences of the truthful voters. While these assumptions are indeed the best possible for the computationally negative results, we note that they are not reflective of typical real-world scenarios. Indeed, concerns regarding privacy of information, and in other cases, the sheer volume of information, would be significant hurdles for manipulators to obtain complete information. Motivated by this, we consider the manipulation problem in a natural *partial information* setting. In particular, we model the partial information of the manipulators about the votes of the non-manipulators as partial orders over the set of candidates. A partial order over the set of candidates will be called a partial vote. Our results show that several of the voting rules that are easy to manipulate in the complete information setting become intractable when the manipulators know only partial votes. Indeed, for many voting rules, we show that even if the ordering of a small number of pairs of candidates is missing from the profile, manipulation becomes an intractable problem. Our results therefore strengthen the view that manipulation may not be practical if we limit the information the manipulators have at their disposal about the votes of other voters [Conitzer *et al.*, 2011].

We introduce three new computational problems that, in a natural way, extend the question of manipulation to the partial information setting. In these problems, the input is a set of partial votes $\mathcal{P}$ corresponding to the votes of the non-manipulators, a non-empty set of manipulators M, and a preferred candidate c. The task in the WEAK MANIPULATION (WM) problem is to determine if there is a way of casting the manipulators' votes such that c wins the election for at least one extension of the partial votes in $\mathcal{P}$. On the other hand, in the STRONG MANIPULATION (SM) problem, we would like to know if there is a way of casting the manipulators' votes such that c wins the election in *every extension* of the partial votes in $\mathcal{P}$.

We also introduce the problem of OPPORTUNISTIC MANIPULATION (OM), which is an "intermediate" notion of manipulation. Let us call an extension of a partial profile *viable* if it is possible for the manipulators to vote in such a way that the manipulators' desired candidate wins in that extension. In other words, a viable extension is a YES-instance of the standard CM problem. We have an opportunistic manipulation when it is possible for the manipulators to cast a vote which makes c win the election in all viable extensions. Note that any YES-instance of SM is also an YES-instance of OM, but this may not be true in reverse. As a particularly extreme example, consider a partial profile where there are no viable extensions: this would be a NO-instance for SM, but a (vacuous) YES-instance of OM. The OM problem allows us to explore a more relaxed notion of manipulation: one where the manipulators are obliged to be successful only in extensions where it is possible to be successful. Note that the goal with SM is to be successful in all extensions, and therefore the only interesting instances are the ones where all extensions are viable.

	$A \succ B \succ C$ $A \succ B$	
Extension 1	Extension 2	Extension 3
$A \succ B \succ C$ $A \succ B \succ C$	$A \succ B \succ C$ $A \succ C \succ B$	$A \succ B \succ C$ $C \succ A \succ B$

Table 1: An example of a partial profile. Consider the plurality voting rule with one manipulator. If the favorite candidate is A, then the manipulator simply has to place A on the top his vote to make A win in *any* extension. If the favorite candidate is B, there is *no vote* that makes B win in any extension. Finally, if the favorite candidate is C, then with a vote that places C on top, the manipulator can make C win in the only viable extension (Extension 2).

It is easy to see that YES instance of SM is also a YES instance of OM and WM. Beyond this, we remark that all the three problems are questions with different goals, and neither of them render the other redundant. We refer the reader to Section 1.2 for a simple example distinguishing these scenarios.

All the problems above generalize CM, and hence any computational intractability result for CM immediately yields a corresponding intractability result for WM, SM, and OM under the same setting. For example, it is known that the CM problem is intractable for the maximin voting rule when we have at least two manipulators [Xia *et al.*, 2009]. Hence, the WM, SM, and OM problems are intractable for the maximin voting rule when we have at least two manipulators.

1.3 Related Work and Our Contributions

A notion of manipulation under partial information has been considered by [Conitzer *et al.*, 2011]. However, they focus on whether or not there exists a dominating manipulation and show that it is NP-hard for many common voting rules. Given some partial votes, a dominating manipulation is a non-truthful vote that the manipulator can cast which makes the winner at least as preferable (and sometimes more preferable) as the winner when the manipulator votes truthfully. The dominating manipulation problem and the WM, OM, and SM problems do not seem to have any apparent complexity-theoretic connection. For example, the dominating manipulation problem is NP-hard for all the common voting rules except plurality and veto, whereas, the SM problem is easy for most of the cases (see Table 2). However, the results in [Conitzer *et al.*, 2011] establish the fact that it is indeed possible to make manipulation intractable by restricting the amount of information the manipulators possess about the votes of the other voters. Elkind and Erdélyi [Elkind and Erdélyi, 2012] study manipulation under voting rule uncertainty. However, in our work, the voting rule is fixed and known to the manipulators.

Two closely related problems that have been extensively studied in the context of incomplete votes are POSSIBLE WINNER (PW) and NECESSARY WINNER (NW) [Konczak and Lang, 2005]. In the PW problem, we are given a set of partial votes $\mathcal{P}$ and a candidate c, and the question is whether

<table>
<tr><th></th><th>WM, $\ell = 1$</th><th>WM</th><th>OM, $\ell = 1$</th><th>OM</th><th>SM, $\ell = 1$</th><th>SM</th></tr>
<tr><td>Plurality</td><td colspan="2" rowspan="2">P</td><td colspan="4" rowspan="2">P</td></tr>
<tr><td>Veto</td></tr>
<tr><td>k-Approval</td><td colspan="2" rowspan="4">NP-complete</td><td colspan="2" rowspan="3">coNP-hard</td><td colspan="2" rowspan="4">P</td></tr>
<tr><td>k-Veto</td></tr>
<tr><td>Bucklin</td></tr>
<tr><td>Fallback</td><td colspan="2">P</td></tr>
<tr><td>Borda</td><td colspan="2" rowspan="3">NP-complete</td><td colspan="2" rowspan="3">coNP-hard</td><td rowspan="2">P</td><td rowspan="3">NP-hard</td></tr>
<tr><td>maximin</td></tr>
<tr><td>Copeland$^\alpha$</td><td>coNP-hard</td></tr>
</table>

Table 2: Summary of Results (ℓ denotes the number of manipulators). The results in white follow immediately from the literature (Observation 1 to 3)

there exists an extension of $\mathcal{P}$ where c wins, while in the NW problem, the question is whether c is a winner in every extension of $\mathcal{P}$. Following the work of [Konczak and Lang, 2005], a number of special cases and variants of the PW problem have been studied in the literature [Gaspers *et al.*, 2014, and references therein]. The flavor of the WM problem is clearly similar to PW. However, we emphasize that there are subtle distinctions between the two problems. A more elaborate comparison is made in the next section.

Our primary contribution in this paper is to propose and study three natural and realistic generalizations of the computational problem of manipulation. Our results show that many voting rules that are vulnerable to manipulation in the complete information setting become resistant to manipulation in the settings that we study. We summarize the complexity results in this paper in Table 2.

2 Preliminaries

In this section, we begin by providing the technical definitions and notations that we will need in the subsequent sections. We refer the reader to [Brandt *et al.*, 2015] for a more comprehensive summary of voting rules — we only define, for completeness, the ones that we use in the technical sections. We then formulate the problems that capture our notions of manipulation when the votes are given as partial orders, and finally draw comparisons with related problems that are already studied in the literature of computational social choice theory.

2.1 Notations and Definitions

Let $\mathcal{V} = \{v_1, \dots, v_n\}$ be the set of all *voters* and $\mathcal{C} = \{c_1, \dots, c_m\}$ the set of all *candidates*. If not specified explicitly, n and m denote the total number of voters and the total number of candidates respectively. Each voter v_i's *vote* is a *preference* $\succ_i$ over the candidates which is a linear order over $\mathcal{C}$. We denote the set of all linear orders over $\mathcal{C}$ by $\mathcal{L}(\mathcal{C})$. Hence, $\mathcal{L}(\mathcal{C})^n$ denotes the set of all n-voters' preference profile $(\succ_1, \dots, \succ_n)$. A map $r : \cup_{n,|\mathcal{C}| \in \mathbb{N}^+} \mathcal{L}(\mathcal{C})^n \longrightarrow 2^{\mathcal{C}} \setminus \{\emptyset\}$ is called a *voting rule*. For some preference profile $\succ \in \mathcal{L}(\mathcal{C})^n$, if $r(\succ) = \{w\}$, then we say w wins uniquely and we write $r(\succ) = w$. From here on, whenever we say some candidate w wins, we mean that the candidate w wins uniquely. For simplicity, we restrict ourselves to the unique winner case in this paper. All our proofs can be easily extended for the co-winner case. A more general setting is an *election* where the votes are only *partial orders* over candidates. A *partial order* is a relation that is *reflexive, antisymmetric*, and *transitive*. A partial vote can be extended to possibly more than one linear vote depending on how we fix the order of the unspecified pairs of candidates. For example, in an election with the set of candidates $\mathcal{C} = \{a, b, c\}$, a valid partial vote can be $a \succ b$. This partial vote can be extended to three linear votes namely, $a \succ b \succ c$, $a \succ c \succ b$, $c \succ a \succ b$. In this paper, we often define a partial vote like $\succ \setminus A$, where $\succ \in \mathcal{L}(\mathcal{C})$ and $A \subset \mathcal{C} \times \mathcal{C}$, by which we mean the partial vote obtained by removing the order among the pair of candidates in A from $\succ$. Also, whenever we do not specify the order among a set of candidates while describing a complete vote, the statement/proof is correct in whichever way we fix the order among them. We now give examples of some common voting rules.

k-Approval. The k-approval score of a candidate x is the number of votes where x is placed within the top k positions and the winners are the candidates with maximum k-approval score.

Bucklin and simplified Bucklin. Let ℓ be the minimum integer such that at least one candidate gets majority within top ℓ positions of the votes. The winners under the simplified Bucklin voting rule are the candidates having more than $n/2$ votes within top ℓ positions. The winners under the Bucklin voting rule are the candidates appearing within top ℓ positions of the votes highest number of times.

2.2 Problem Definitions

We now formally define the three problems that we consider in this work, namely Weak Manipulation, Opportunistic Manipulation, and Strong Manipulation. Let r be a fixed voting rule. We first introduce the problem of Weak Manipulation.

Definition 1. r**-Weak Manipulation** (WM)
Given a set of partial votes $\mathcal{P}$ over a set of candidates $\mathcal{C}$, a positive integer ℓ (> 0) denoting the number of manipulators, and a candidate c, do there exist votes $\succ_1, \dots, \succ_\ell \in \mathcal{L}(\mathcal{C})$ such that there exists an extension $\succ \in \mathcal{L}(\mathcal{C})^{|\mathcal{P}|}$ of $\mathcal{P}$ with $r(\succ, \succ_1, \dots, \succ_\ell) = c$?

To define Opportunistic Manipulation, we first introduce the notion of an (r, c)-Opportunistic Voting Profile, where r is a voting rule and c is any particular candidate.

Definition 2. (r, c)***-Opportunistic Voting Profile***
Let ℓ be the number of manipulators and $\mathcal{P}$ a set of partial votes. An ℓ-voter profile $(\succ_i)_{i \in [\ell]} \in \mathcal{L}(\mathcal{C})^\ell$ is called (r, c)-opportunistic if for each extension $\overline{\mathcal{P}}$ of $\mathcal{P}$ for which there exists an ℓ-vote profile $(\succ'_i)_{i \in [\ell]} \in \mathcal{L}(\mathcal{C})^\ell$ with $r(\overline{\mathcal{P}} \cup (\succ'_i)_{i \in [\ell]}) = c$, we have $r(\overline{\mathcal{P}} \cup (\succ_i)_{i \in [\ell]}) = c$.

In other words, an ℓ-vote profile is (r, c)-opportunistic with respect to a partial profile if, when put together with the truthful votes of any extension, c wins if the extension is viable to begin with. We are now ready to define OPPORTUNISTIC MANIPULATION.

Definition 3. r**-OPPORTUNISTIC MANIPULATION** (OM)
Given a set of partial votes $\mathcal{P}$ over a set of candidates $\mathcal{C}$, a positive integer ℓ (> 0) denoting the number of manipulators, and a candidate c, does there exist an (r, c)-opportunistic ℓ-vote profile?

We finally define the STRONG MANIPULATION problem.

Definition 4. r**-STRONG MANIPULATION** (SM)
Given a set of partial votes $\mathcal{P}$ over a set of candidates $\mathcal{C}$, a positive integer ℓ (> 0) denoting the number of manipulators, and a candidate c, do there exist votes $(\succ_i)_{i\in\ell} \in \mathcal{L}(\mathcal{C})$ such that for every extension $\succ \in \mathcal{L}(\mathcal{C})^{|\mathcal{P}|}$ of $\mathcal{P}$, we have $r(\succ, (\succ_i)_{i\in[\ell]}) = c$?

We use $(\mathcal{P}, \ell, c)$ to denote instances of WM, OM, and SM, where $\mathcal{P}$ denotes the profile of partial votes, ℓ denotes the number of manipulators, and c denotes the desired winner.

2.3 Comparison with PW and CM.

For any fixed voting rule, the WM problem with ℓ manipulators reduces to the PW problem. This is achieved by simply using the same set as truthful votes and introducing ℓ empty votes. We summarize this in the observation below.

Observation 1. *The* WM *problem many-to-one reduces to the* PW *problem for every voting rule.*

However, whether the PW problem reduces to the WM problem or not is not clear since in any WM problem instance, there must exist at least one manipulator and a PW instance may have no empty vote. From a technical point of view, the difference between the WM and PW problems may look marginal; however we believe that the WM problem is a very natural generalization of the CM problem in the partial information setting and thus worth studying. Similarly, it is easy to show, that the CM problem with ℓ manipulators reduces to WM, OM, and SM problems with ℓ manipulators, since the former is a special case of the latter ones.

Observation 2. *The* CM *problem with ℓ manipulators many-to-one reduces to* WM*,* OM*, and* SM *problems with ℓ manipulators for all voting rules and for all positive integers ℓ.*

Finally, we note that the CM problem with ℓ manipulators can be reduced to the WM problem with just one manipulator, by introducing $\ell - 1$ empty votes. These votes can be used to witness a good extension in the forward direction. In the reverse direction, given an extension where the manipulator is successful, the extension can used as the manipulator's votes. This argument leads to the following.

Observation 3. *The* CM *problem with ℓ manipulators many-to-one reduces to the* WM *problem with one manipulator for every voting rule and for every positive integer ℓ.*

This observation can be used to derive the hardness of WM for even one manipulator whenever the hardness for CM is known for any fixed number of manipulators (for instance, this is the case for the voting rules such as Borda, maximin and Copeland). However, determining the complexity of WM with one manipulator requires further work for voting rules where CM is polynomially solvable for any number of manipulators (such as k-approval, Plurality, and so on).

3 Hardness Results

In this section, we provide an overview of our hardness results. While some of our reductions are from the POSSIBLE WINNER problem, the other reductions in this section are from the EXACT COVER BY 3-SETS problem, also referred to as X3C. This is a well-known NP-complete [Garey and Johnson, 1979] problem, and is defined as follows.

Definition 5 (Exact Cover by 3-Sets (X3C))**.** *Given a set $\mathcal{U}$ and a collection $\mathcal{S} = \{S_1, S_2, \ldots, S_t\}$ of t subsets of $\mathcal{U}$ with $|S_i| = 3\ \forall i = 1, \ldots, t$, does there exist a $\mathcal{T} \subset \mathcal{S}$ with $|\mathcal{T}| = \frac{|\mathcal{U}|}{3}$ such that $\cup_{X\in\mathcal{T}} X = \mathcal{U}$?*

We use $\overline{\text{X3C}}$ to refer to the complement of X3C, which is to say that an instance of $\overline{\text{X3C}}$ is a YES instance if and only if it is a NO instance of X3C. Due to constraints of space, we defer the details of all the proofs to full version of the paper [Dey *et al.*,]. We provide the details of one of our reductions, whose style is representative of our general approach in the other cases as well. The rest of this section is organized according to the problems being addressed.

Weak Manipulation. To begin with, recall that the CM problem is NP-complete for the Borda [Davies *et al.*, 2011; Betzler *et al.*, 2011], maximin [Xia *et al.*, 2009], and Copeland$^\alpha$ [Faliszewski *et al.*, 2008; 2009; 2010] voting rules for rational $\alpha \in [0, 1] \setminus \{0.5\}$, when we have two manipulators. Therefore, it follows from Observation 3 that the WM problem is NP-complete for the Borda, maximin, and Copeland$^\alpha$ voting rules for rational $\alpha \in [0, 1] \setminus \{0.5\}$, even with one manipulator.

For the k-approval and k-veto voting rules, we reduce from the corresponding PW problems. While it is natural to start from the same voting profile, the main challenge is in undoing the advantage that the favorite candidate receives from the manipulator's vote, in the reverse direction. We also prove that the WEAK MANIPULATION problem for the Bucklin and simplified Bucklin rules is NP-complete, by a reduction from X3C. Our reduction is along the lines of the reduction given in [Xia and Conitzer, 2011] (which was for the simplified Bucklin voting rule).

Strong Manipulation. We know that the CM problem is NP-complete for the Borda, maximin, and Copeland$^\alpha$ voting rules for rational $\alpha \in [0, 1] \setminus \{0.5\}$, when we have two manipulators. Thus, it follows from Observation 2 that SM is NP-hard for Borda, maximin, and Copeland$^\alpha$ voting rules for rational $\alpha \in [0, 1] \setminus \{0.5\}$ for at least two manipulators.

For the case of one manipulator, SM turns out to be polynomial-time solvable for most other voting rules. For Copeland$^\alpha$, however, we show that the problem is co-NP-hard for $\alpha \in [0, 1]$ for a single manipulator, even when the number of undetermined pairs in each vote is bounded by a constant. This is achieved by a careful reduction from $\overline{\text{X3C}}$.

Opportunistic Manipulation. All our reductions for the co-NP-hardness for OM start from $\overline{\text{X3C}}$. We note that all our hardness results hold even when there is only one manipulator. Our overall approach is the following. We engineer a set of partial votes in such a way that the manipulator is forced to vote in a limited number of ways to have any hope of making her favorite candidate win. For each such vote, we demonstrate a viable extension where the vote fails to make the candidate a winner, leading to a NO instance of OM. These extensions rely on the existence of an exact cover. On the other hand, we show that if there is no set cover, then there is no viable extension, thereby leading to an instance that is vacuously a YES instance of OM.

We provide one of the reductions below to convey a flavor of the techniques involved. The constructions for the other voting rules are in a similar spirit, but we remark that the details are typically more involved.

Theorem 1. *The* OM *problem is* co-NP*-hard for the k-approval voting rule for constant $k \geqslant 3$ even when the number of manipulators is one and the number of undetermined pairs in each vote is no more than* 15.

Proof: We reduce $\overline{\text{X3C}}$ to OM for k-approval rule. Let $(\mathcal{U} = \{u_1, \ldots, u_m\}, \mathcal{S} = \{S_1, S_2, \ldots, S_t\})$ is an $\overline{\text{X3C}}$ instance. We construct a corresponding OM instance for k-approval voting rule as follows. We begin by introducing a candidate for every element of the universe, along with $k-3$ dummy candidates (denoted by $\mathcal{W}$), and special candidates $\{c, z_1, z_2, d, x, y\}$. Formally, we have:

$$\text{Candidate set } \mathcal{C} = \mathcal{U} \cup \{c, z_1, z_2, d, x, y\} \cup \mathcal{W}.$$

Now, for every set S_i in the universe, we define the following total order on the candidate set, which we denote by $\mathcal{P}'_i$:

$$\mathcal{W} \succ S_i \succ y \succ z_1 \succ z_2 \succ x \succ (\mathcal{U} \setminus S_i) \succ c \succ d$$

Using $\mathcal{P}'_i$, we define the partial vote $\mathcal{P}_i$ as follows:

$$\mathcal{P}_i = \mathcal{P}'_i \setminus (\{\{y, x, z_1, z_2\} \times S_i\} \cup \{(z_1, z_2), (x, z_1), (x, z_2)\}).$$

We denote the set of partial votes $\{\mathcal{P}_i : i \in [t]\}$ by $\mathcal{P}$ and $\{\mathcal{P}'_i : i \in [t]\}$ by $\mathcal{P}'$. We remark that the number of undetermined pairs in each partial vote $\mathcal{P}_i$ is 15.

We now invoke Lemma 1 from [Dey *et al.*, 2015b], which allows to achieve any pre-defined scores on the candidates using only polynomially many additional votes. Using this, we add a set $\mathcal{Q}$ of complete votes with $|\mathcal{Q}| = \text{poly}(m, t)$ to ensure the following scores, where we denote the k-approval score of a candidate from a set of votes $\mathcal{V}$ by $s_{\mathcal{V}}(\cdot)$: $s_{\mathcal{Q}}(z_1) = s_{\mathcal{Q}}(z_2) = s_{\mathcal{Q}}(y) = s_{\mathcal{Q}}(c) - {}^{m}/_{3}; s_{\mathcal{Q}}(d), s_{\mathcal{Q}}(w) \leqslant s_{\mathcal{Q}}(c) - 2t \;\forall w \in \mathcal{W}; s_{\mathcal{Q}}(x) = s_{\mathcal{Q}}(c) - 1; s_{\mathcal{P}' \cup \mathcal{Q}}(u_j) = s_{\mathcal{Q}}(c) + 1 \;\forall j \in [m]$.

Our reduced instance is $(\mathcal{P} \cup \mathcal{Q}, 1, c)$. We first argue that if we had a YES instance of $\overline{\text{X3C}}$ (in other words, there is no exact cover), then we have a YES instance of OM. It turns out that this will follow from the fact that there are no viable extensions, because, as we will show next, a viable extension implies the existence of an exact set cover.

To this end, first observe that the partial votes are constructed in such a way that c gets no additional score from *any* extension. Assuming that the manipulator approves c (without loss of generality), the final score of c in any extension is going to be $s_{\mathcal{Q}}(c) + 1$. Now, in any viable extension, every candidate u_j has to be "pushed out" of the top k positions at least once. Observe that whenever this happens, y is forced into the top k positions. Since y is behind the score of c by only $m/3$ votes, S_i's can be pushed out of place in only $m/3$ votes. For every u_j to lose one point, these votes must correspond to an exact cover. Therefore, if there is no exact cover, then there is no viable extension, showing one direction of the reduction.

On the other hand, suppose we have a NO instance of $\overline{\text{X3C}}$ – that is, there is an exact cover. We will now use the exact cover to come up with two viable extensions, both of which require the manipulator to vote in different ways to make c win. Therefore, there is no single manipulative vote that accounts for both extensions, leading us to a NO instance of OM.

First, consider this completion of the partial votes:

$$i = 1, \mathcal{W} \succ y \succ x \succ z_1 \succ z_2 \succ S_i \succ (\mathcal{U} \setminus S_i) \succ c \succ d$$

$$2 \leqslant i \leqslant {}^{m}/_{3}, \mathcal{W} \succ y \succ z_1 \succ z_2 \succ x \succ S_i \succ (\mathcal{U} \setminus S_i) \succ c \succ d$$

$${}^{m}/_{3} + 1 \leqslant i \leqslant t, \mathcal{W} \succ S_i \succ y \succ z_1 \succ z_2 \succ x \succ (\mathcal{U} \setminus S_i) \succ c \succ d$$

Notice that in this completion, once accounted for along with the votes in $\mathcal{Q}$, the score of c is tied with the scores of all u_j's, z_1, x and y, while the score of z_2 is one less than the score of c. Therefore, the only k candidates that the manipulator can afford to approve are $\mathcal{W}$, the candidates c, d and z_2. However, consider the extension that is identical to the above except with the first vote changed to:

$$\mathcal{W} \succ y \succ x \succ z_2 \succ z_1 \succ S_i \succ (\mathcal{U} \setminus S_i) \succ c \succ d$$

Here, on the other hand, the only way for c to be an unique winner is if the manipulator approves $\mathcal{W}, c, d$ and z_1. Therefore, it is clear that there is no way for the manipulator to provide a consolidated vote for both these profiles. Therefore, we have a NO instance of OM. ☐

4 Polynomial Time Algorithms

We now turn to the polynomial time cases depicted in Table 2. As in the previous section, we will provide one of our proofs in detail, which is nonetheless representative of the overall flavor of the arguments for the other cases. This section is organized in three parts, one for each problem considered.

Weak Manipulation. Since the PW problem is in P for the plurality and the veto voting rules [Betzler and Dorn, 2010], it follows from Observation 1 that the WM problem is in P for the plurality and veto voting rules for any number of manipulators.

Strong Manipulation. We now discuss the SM problem. The common flavor in all our algorithms is the following: we try to devise an extension that is as adversarial as possible for the favorite candidate c, and if we can make c win in such an extension, then roughly speaking, such a strategy should work for other extensions as well (where the situation only improves for c). However, it is challenging to come up with an extension that is globally dominant over all the others in the sense that we just described. So what we do instead is we consider every potential nemesis w who might win instead of c, and we build profiles that are "as good as possible" for w and "as bad as possible" for c. Each such profile leads us to constraints on how much the manipulators can afford to favor w (in terms of which positions among the manipulative votes are *safe* for w). We then typically show that we can determine whether there exists a set of votes that respects these constraints, either by using a greedy strategy or by an appropriate reduction to a flow problem. We note that the overall spirit here is similar to the approaches commonly used for solving NW problems, but as we will see, there are non-trivial differences in the details. We provide an exposition of our ideas for the case of the simplified Bucklin voting rule. We also note that the proof is quite similar for the Bucklin, Fallback, and simplified Fallback voting rules.

Theorem 2. *The* SM *problem is in* P *for the simplified Bucklin voting rules, for any number of manipulators.*

Proof: Let $(\mathcal{C}, \mathcal{P}, M, c)$ be an instance of SM for simplified Bucklin, and let m denote the total number of candidates in this instance. Recall that the manipulators have to cast their votes so as to ensure that the candidate c wins in every possible extension of $\mathcal{P}$. We use $\mathcal{Q}$ to denote the set of manipulating votes that we will construct. To begin with, without loss of generality, the manipulators place c in the top position of all their votes. We now have to organize the positioning of the remaining candidates across the votes of the manipulators to ensure that c is a necessary winner of the profile $(\mathcal{P}, \mathcal{Q})$.

To this end, we would like to develop a system of constraints indicating the overall number of times that we are free to place a candidate $w \in \mathcal{C} \setminus \{c\}$ among the top ℓ positions in the profile $\mathcal{Q}$. In particular, let us fix $w \in \mathcal{C} \setminus \{c\}$ and $2 \leqslant \ell \leqslant m$. Let $\eta_{w,\ell}$ be the maximum number of votes of $\mathcal{Q}$ in which w can appear in the top ℓ positions. Our first step is to compute necessary conditions for $\eta_{w,\ell}$.

We use $\overline{\mathcal{P}}_{w,\ell}$ to denote a set of complete votes that we will construct based on the given partial votes. Intuitively, these votes will represent the "worst" possible extensions from the point of view of c when pitted against w. These votes are engineered to ensure that the manipulators can make c win the elections $\overline{\mathcal{P}}_{w,\ell}$ for all $w \in \mathcal{C} \setminus \{c\}$ and $\ell \in \{2, \ldots, m\}$, if, and only if, they can strongly manipulate in favor of c. More formally, there exists a voting profile $\mathcal{Q}$ of the manipulators so that c wins the election $\overline{\mathcal{P}}_{w,\ell} \cup \mathcal{Q}$, for all $w \in \mathcal{C} \setminus \{c\}$ and $\ell \in \{2, \ldots, m\}$ if and only if c wins in every extension of the profile $\mathcal{P} \cup \mathcal{Q}$.

We now describe the profile $\overline{\mathcal{P}}_{w,\ell}$. The construction is based on the following case analysis, where our goal is to ensure that, to the extent possible, we position c out of the top $\ell - 1$ positions, and incorporate w among the top ℓ positions.

– Let $v \in \mathcal{P}$ be such that either c and w are incomparable or $w \succ c$. We add the complete vote v' to $\overline{\mathcal{P}}_{w,\ell}$, where v' is obtained from v by placing w at the highest possible position and c at the lowest possible position, and extending the remaining vote arbitrarily.

– Let $v \in \mathcal{P}$ be such that $c \succ w$, but there are at least ℓ candidates that are preferred over w in v. We add the complete vote v' to $\overline{\mathcal{P}}_{w,\ell}$, where v' is obtained from v by placing c at the lowest possible position, and extending the remaining vote arbitrarily.

– Let $v \in \mathcal{P}$ be such that c is forced to be within the top $\ell - 1$ positions, then we add the complete vote v' to $\overline{\mathcal{P}}_{w,\ell}$, where v' is obtained from v by first placing w at the highest possible position followed by placing c at the lowest possible position, and extending the remaining vote arbitrarily.

– In the remaining votes, notice that whenever w is in the top ℓ positions, c is also in the top $\ell - 1$ positions. Let $\mathcal{P}^*_{w,\ell}$ denote this set of votes, and let t be the number of votes in $\mathcal{P}^*_{w,\ell}$.

The rest of the proof goes via case analysis on the number of times c is placed in the top $\ell - 1$ positions in the profile $\overline{\mathcal{P}}_{w,\ell} \cup \mathcal{Q}$, and the number of times w is placed in the top ℓ positions in the profile $\overline{\mathcal{P}}_{w,\ell}$. We defer the detailed proof to a full version of this paper. □

Opportunistic Manipulation. For plurality, Fallback, and simplified Fallback voting rules, it turns out that the voting profile where all manipulators approve only c is a c-opportunistic voting profile, and therefore it is easy to devise a manipulative vote. For Veto, however, a more intricate argument is involved, that requires building a system of constraints and a reduction to a suitable instance of maxflow. We defer the details to a full version of this paper.

5 Conclusion

We present a fresh perspective on the use of computational complexity as a barrier to manipulation, particularly in cases that were thought to be dead-ends (because the traditional manipulation problem was polynomially solvable). Our work is likely to be the starting point for further explorations: other kinds election control in partial information setting, average case analysis of manipulation with partial information etc.

Acknowledgement. Palash Dey wishes to gratefully acknowledge support from Google India for providing him with a special fellowship for carrying out his doctoral work. Neeldhara Misra acknowledges support by the INSPIRE Faculty Scheme, DST India (project IFA12-ENG-31).

References

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

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

[Betzler and Dorn, 2010] Nadja Betzler and Britta Dorn. Towards a dichotomy for the possible winner problem in elections based on scoring rules. *J. Comput. Syst. Sci.*, 76(8):812–836, 2010.

[Betzler *et al.*, 2011] Nadja Betzler, Rolf Niedermeier, and Gerhard J Woeginger. Unweighted coalitional manipulation under the Borda rule is NP-hard. In *IJCAI*, volume 11, pages 55–60, 2011.

[Brandt *et al.*, 2015] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel Procaccia. Handbook of computational social choice, 2015.

[Conitzer *et al.*, 2007] Vincent Conitzer, Tuomas Sandholm, and Jérôme Lang. When are elections with few candidates hard to manipulate? *Journal of the ACM (JACM)*, 54(3):14, 2007.

[Conitzer *et al.*, 2011] Vincent Conitzer, Toby Walsh, and Lirong Xia. Dominating manipulations in voting with partial information. In *International Conference on Artificial Intelligence (AAAI)*, volume 11, pages 638–643, 2011.

[Davies *et al.*, 2011] J. Davies, G. Katsirelos, N. Narodytska, and T. Walsh. Complexity of and algorithms for Borda manipulation. In *Proceedings of the International Conference on Artificial Intelligence (AAAI)*, pages 657–662, 2011.

[Dey and Narahari, 2015] Palash Dey and Y Narahari. Asymptotic collusion-proofness of voting rules: The case of large number of candidates. *Studies in Microeconomics*, 3(2):120–139, 2015.

[Dey *et al.*,] P. Dey, N. Misra, and Y. Narahari. Complexity of Manipulation with Partial Information in Voting. *ArXiv e-prints*.

[Dey *et al.*, 2015a] Palash Dey, Neeldhara Misra, and Y Narahari. Detecting possible manipulators in elections. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 1441–1450. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

[Dey *et al.*, 2015b] Palash Dey, Neeldhara Misra, and Y Narahari. Kernelization complexity of possible winner and coalitional manipulation problems in voting. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 87–96. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

[Dey *et al.*, 2016] Palash Dey, Neeldhara Misra, and Y Narahari. Kernelization complexity of possible winner and coalitional manipulation problems in voting. *Theoretical Computer Science*, 616:111–125, 2016.

[Dey, 2015] Palash Dey. Computational complexity of fundamental problems in social choice theory. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 1973–1974. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

[Elkind and Erdélyi, 2012] Edith Elkind and Gábor Erdélyi. Manipulation under voting rule uncertainty. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 627–634. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[Faliszewski *et al.*, 2008] Piotr Faliszewski, Edith Hemaspaandra, and Henning Schnoor. Copeland voting: Ties matter. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '08, pages 983–990, Richland, SC, 2008. International Foundation for Autonomous Agents and Multiagent Systems.

[Faliszewski *et al.*, 2009] Piotr Faliszewski, Edith Hemaspaandra, Lane A. Hemaspaandra, and Jörg Rothe. Llull and copeland voting computationally resist bribery and constructive control. *Journal of Artificial Intelligence Research (JAIR)*, 35:275–341, 2009.

[Faliszewski *et al.*, 2010] Piotr Faliszewski, Edith Hemaspaandra, and Henning Schnoor. Manipulation of copeland elections. In *9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010), Toronto, Canada, May 10-14, 2010, Volume 1-3*, pages 367–374, 2010.

[Garey and Johnson, 1979] Michael R Garey and David S Johnson. *Computers and Intractability*, volume 174. freeman New York, 1979.

[Gaspers *et al.*, 2014] Serge Gaspers, Victor Naroditskiy, Nina Narodytska, and Toby Walsh. Possible and necessary winner problem in social polls. In *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 613–620. International Foundation for Autonomous Agents and Multiagent Systems, 2014.

[Gibbard, 1973] A. Gibbard. Manipulation of voting schemes: a general result. *Econometrica: Journal of the Econometric Society*, pages 587–601, 1973.

[Isaksson *et al.*, 2012] Marcus Isaksson, Guy Kindler, and Elchanan Mossel. The geometry of manipulationa quantitative proof of the gibbard-satterthwaite theorem. *Combinatorica*, 32(2):221–250, 2012.

[Konczak and Lang, 2005] Kathrin Konczak and Jérôme Lang. Voting procedures with incomplete preferences. In *Proc. International Joint Conference on Artificial Intelligence-05 Multidisciplinary Workshop on Advances in Preference Handling*, volume 20, 2005.

[Procaccia and Rosenschein, 2007a] Ariel D. Procaccia and Jeffrey S. Rosenschein. Average-case tractability of manipulation in voting via the fraction of manipulators. In *6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007), Honolulu, Hawaii, USA, May 14-18, 2007*, page 105, 2007.

[Procaccia and Rosenschein, 2007b] Ariel D. Procaccia and Jeffrey S. Rosenschein. Junta distributions and the average-case complexity of manipulating elections. *Journal of Artificial Intelligence Research*, 28:157–181, February 2007.

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

[Walsh, 2011] Toby Walsh. Where are the hard manipulation problems? *Journal of Artificial Intelligence Research*, pages 1–29, 2011.

[Xia and Conitzer, 2011] Lirong Xia and Vincent Conitzer. Determining possible and necessary winners under common voting rules given partial orders. *Journal of Artificial Intelligence Research (JAIR)*, 41(2):25–67, 2011.

[Xia *et al.*, 2009] Lirong Xia, Michael Zuckerman, Ariel D Procaccia, Vincent Conitzer, and Jeffrey S Rosenschein. Complexity of unweighted coalitional manipulation under some common voting rules. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, volume 9, pages 348–353, 2009.