Coalitional Voting Manipulation: A Game-Theoretic Perspective

Yoram Bachrach
Microsoft Research
Cambridge, UK

Edith Elkind
School of Physical and
Mathematical Sciences
Nanyang Technological University
Singapore

Piotr Faliszewski
AGH University
of Science and Technology
Kraków, Poland

Abstract

Computational social choice literature has successfully studied the complexity of manipulation in various voting systems. However, the existing models of coalitional manipulation view the manipulating coalition as an exogenous input, ignoring the question of the coalition formation process. While such analysis is useful as a first approximation, a richer framework is required to model voting manipulation in the real world more accurately, and, in particular, to explain how a manipulating coalition arises and chooses its action. In this paper, we apply tools from cooperative game theory to develop a model that considers the coalition formation process and determines which coalitions are likely to form and what actions they are likely to take. We explore the computational complexity of several standard coalitional game theory solution concepts in our setting, and study the relationship between our model and the classic coalitional manipulation problem as well as the now-standard bribery model.

1 Introduction

Voting is a standard method of preference aggregation in multi-agent environments. It allows the agents (voters) to make joint decisions by selecting the most suitable alternative from a given set. However, in settings where voters are selfish and aim to optimize their individual utility, voting suffers from a serious problem: essentially all voting rules are manipulable, i.e., a voter may benefit from misrepresenting her preferences over the alternatives [Gibbard, 1973; Satterthwaite, 1975]. Consequently, classifying voting rules according to their resistance to manipulation has been an active research topic in the last decade (see [Faliszewski and Procaccia, 2010] for an overview)

While the possibility of manipulation by a single voter presents a grave concern from a theoretical perspective, in real-life elections this issue does not usually play a significant role: typically, the outcome of a popular vote is not close enough to be influenced by a single voter. Indeed, a more significant problem is that of coalitional manipulation, where a group of voters coordinates their actions in order to affect the election outcome. The problem of coalitional manipulation was first explicitly introduced by [Conitzer *et al.*, 2007], where the authors also initiated its analysis from the computational perspective. Since then, a number of results on the computational complexity of coalitional manipulation for a variety of voting rules have been obtained (see the Related Work section below).

However, the model of coalitional manipulation proposed in [Conitzer *et al.*, 2007] abstracts away some of the issues that are crucial for realistic modeling of coalitional manipulation scenarios. Specifically, this model assumes that the set of all voters is partitioned into two groups: the honest voters and the manipulators. The honest voters have preferences over the candidates, while the manipulators are single-minded: they simply want to get a specific candidate elected. Thus, the set of manipulators is an *exogenous* variable, given as a part of the input. This definition does not *explain* how the manipulating coalition forms or how it decides which candidate to promote. Arguably, this provides an overly simplistic view of reality: it is natural to expect that the would-be manipulators start out by having preferences over the entire set of candidates, but then decide to cooperate with each other, as they are not satisfied with the outcome of truthful voting.

Against this background, our goal in this paper is to provide an *endogenous* model of coalitional manipulation that is based on coalitional game theory. We assume that all agents have preferences over the set of candidates; we make the standard assumption that these preferences are common knowledge. In addition, a subset of the agents are strategic and would consider forming a manipulating coalition if they can profit by doing so. Given this setup, each voting rule induces a coalitional game, where the players are the strategic agents (we will refer to them as *colluders*), and the set of outcomes that are feasible for a coalition is determined by the set of candidates that the players in that coalition can turn into election winners. We consider a *transferable utility* model, where the colluders have comparable utilities (given in a "common currency") for each candidate and that they can commit to making payments to each other.

We study several natural computational problems regarding the coalitional game induced by the voting domain, such as finding the optimal action a coalition can take, identifying coalitions whose optimal action is to support a certain candidate, calculating a player's power in the game and testing whether an outcome is in the core. While exploring these issues, we also examine the relation between our model and the classic coalitional manipulation model and the voting bribery model [Faliszewski *et al.*, 2009]. Our contributions fall into two main categories. First, we introduce a cooperative game-theoretic model of voting manipulation, and study the complexity of natural solution concepts in this model. Second, and on a more fundamental level, even though our work is motivated by a critique of the standard framework of voting manipulation, we show that many classic computational social choice results have natural interpretations in our game-theoretic model. For example, results on the complexity of coalitional manipulation—as defined in [Conitzer *et al.*, 2007]—translate into results on the complexity of computing coalition values, and results on the complexity of bribery—as defined in [Faliszewski *et al.*, 2009]—translate into, e.g., results on the complexity of testing whether a given coalition is stable. We believe that our model is a useful formalism that captures many aspects of coalition formation in voting.

Related work The complexity of coalitional manipulation, as defined in [Conitzer *et al.*, 2007], received a lot of attention in the recent literature (see, e.g., [Hemaspaandra and Hemaspaandra, 2007; Faliszewski *et al.*, 2010; Xia *et al.*, 2009; Walsh, 2009; Xia *et al.*, 2010]; a more exhaustive list is provided in [Faliszewski and Procaccia, 2010]). However, none of these papers discusses the issue of manipulating coalition formation. A recently proposed model of safe strategic voting [Slinko and White, 2008] addresses this issue using an approach that is different from ours: specifically, under this model a single voter announces a manipulative vote, and may be followed by other voters with the same preferences (see also [Hazon and Elkind, 2010] for the algorithmic analysis and extensions of the model of [Slinko and White, 2008]). While the approach of [Slinko and White, 2008] is more suitable when it is difficult for the manipulators to coordinate, our model is more appropriate when coordination is not an issue; thus, the two approaches complement each other. There is also a number of very recent papers that analyze strategic behavior in voting using the tools of *non-cooperative* game theory (see [Desmedt and Elkind, 2010; Xia and Conitzer, 2010; Meir *et al.*, 2010] and the references therein).

Organization of the paper The paper is organized as follows. Section 2 provides background on (computational) social choice and coalitional game theory. In Section 3 we formally introduce our model. Section 4 focuses on the problem of computing coalitional values. In Section 5 we study the complexity of testing if there is a manipulating coalition supporting a given candidate. Section 6 considers players' power in the game, and, in particular, computing the Shapley values. Section 7 investigates coalitional stability. We conclude in Section 8. We omit most proof due to space constraints.

2 Preliminaries

We write $\mathbb{N} = \{0, 1, 2, \dots\}$, and given a vector $\mathbf{x} \in \mathbb{R}^n$ and a set $S \subseteq \{1, \dots, n\}$, we set $\mathbf{x}(S) = \sum_{i \in S} x_i$.

Voting. An *election* $E = (C, V, \mathcal{P})$ is given by a set $C = \{c_1, \dots, c_m\}$ of *candidates*, a set $V = \{1, \dots, n\}$ of *voters*, and a *preference profile* $\mathcal{P} = (P_1, \dots, P_n)$, where each P_i, $i \in V$, is a linear order over C. The order P_i represents the preferences of the i-th voter; for readability, we sometimes write $\succ_i$ instead of P_i. We denote the set of all linear orders over C by $L(C)$; thus, for any election $E = (C, V, \mathcal{P})$ with $|V| = n$ we have $\mathcal{P} \in L(C)^n$. For any $U \subseteq V$, we write $\mathcal{P}_U = (P_i)_{i \in U}$ and $\mathcal{P}_{-U} = (P_i)_{i \notin U}$; we have $\mathcal{P} = (\mathcal{P}_U, \mathcal{P}_{-U})$.

A *voting rule* $\mathcal{R}$ is a mapping that given an election $E = (C, V, \mathcal{P})$ outputs a candidate $c = \mathcal{R}(E)$, which is called the *winner* of E. When the sets C and V are clear from the context, we will sometimes omit them from the notation and write $\mathcal{R}(\mathcal{P})$ instead of $\mathcal{R}(E)$. Note that we require that each election has a unique winner. Many classic voting rules are, in fact, voting correspondences, i.e., they may output multiple winners. We assume that whenever this is the case, the resulting tie is broken lexicographically. We restrict our attention to voting rules with a poly-time winner determination algorithm.

Manipulation and Bribery. Two well-studied forms of dishonest behavior in elections are *manipulation*, i.e., cheating by voters, and *bribery*, i.e., cheating by an external party that wants to influence the outcome of the election. Below, we define the variants of these problems that are relevant to our work, namely, *coalitional manipulation* and *priced bribery*.

Definition 2.1 ([Conitzer *et al.*, 2007]). *For a voting rule $\mathcal{R}$, an instance $I = (E, S, c)$ of $\mathcal{R}$-*COALITIONAL MANIPULATION *problem is given by an election $E = (C, V, \mathcal{P})$, a set of* manipulators *S, $S \cap V = \emptyset$, and the manipulators' preferred candidate $c \in C$. It is a "yes"-instance if there is a vector $\mathcal{P}_S = (P_i)_{i \in S} \in (L(C))^{|S|}$ such that $\mathcal{R}(\mathcal{P}_{-S}, \mathcal{P}_S) = c$; otherwise, it is a "no"-instance.*

Observe that in the traditional definition of coalitional manipulation the manipulators, unlike honest voters, do not have preferences over the candidates: they simply want to get a particular candidate elected. This definition is convenient because it eliminates the problem of deciding which candidates the manipulators should support.

Definition 2.2 ([Faliszewski *et al.*, 2009]). *For a voting rule $\mathcal{R}$, an instance $I = (E, \mathbf{b}, B, c)$ of $\mathcal{R}$-*\$BRIBERY *problem is given by an election $E = (C, V, \mathcal{P})$ with $|V| = n$, a vector of prices $\mathbf{b} = (b_1, \dots, b_n) \in \mathbb{N}^n$, a budget $B \in \mathbb{N}$, and the briber's preferred candidate $c \in C$. It is a "yes"-instance if there is a vector $\mathcal{P}' = (P'_1, \dots, P'_n)$ over C and a set of voters S such that $P_i = P'_i$ for $i \notin S$, $\mathcal{R}(\mathcal{P}') = c$, and $\mathbf{b}(S) \leq B$; otherwise, it is a "no"-instance.*

We will also consider settings with weighted voters, where each voter $i \in V$ has a non-negative integer weight w_i; we denote the weight vector by $\mathbf{w} = (w_1, \dots, w_n)$. To apply a voting rule $\mathcal{R}$ to a weighted election, we replace each voter i with w_i voters whose preferences are identical to those of i. The definitions of coalitional manipulation and \$bribery can be adapted to this setting in a straightforward manner; in particular, when a voter of weight w_i is bribed or participates in a manipulation, we require that all w_i "copies" of this voter vote in the same way.

Computational Complexity. We assume familiarity with basic notions of computational complexity, such as polynomial-time algorithms and classes NP and coNP. A somewhat less standard notion is that of *strong* NP-hardness: a problem is said to be strongly NP-hard if it remains NP-hard even if all numbers in the input (such as, e.g., bribery prices) are given in unary. A related notion is that of a *pseudopolynomial* algorithm: an algorithm is said to be pseudopolynomial if its running time is polynomial in the numeric value of the input.

Coalitional Games. Coalitional games model settings where players form coalitions and derive benefits from collaboration. We assume *transferable utility* model, that is, the members of a coalition can freely distribute the benefits they obtain by working together. It is convenient to think of these benefits as monetary. Formally, a coalitional game $G = (N, v)$ is given by a set of *players* $N = \{1, \dots, |N|\}$ and a characteristic function $v : 2^N \to \mathbb{R}^+ \cup \{0\}$, which for each *coalition* of players $S \subseteq N$ outputs the total amount of money that the players in S can earn by working together. It is standard to normalize the characteristic function by requiring $v(\emptyset) = 0$. A game is called *monotone* if $v(S) \le v(T)$ for any $S, T \subseteq N$ such that $S \subseteq T$. A player $i \in N$ is called a *dummy* if $v(S) = v(S \setminus \{i\})$ for all $S \subseteq N$.

An *outcome* of a game G is a vector $\mathbf{x} = (x_1, \dots, x_{|N|})$ that satisfies $x_i \ge 0$ for all $i \in N$ and $\mathbf{x}(N) = v(N)$. An outcome $\mathbf{x}$ is said to be *stable* if $\mathbf{x}(S) \ge v(S)$ for any $S \subseteq N$; the set of all stable outcomes of a game is called the *core*.

Another useful solution concept is that of Shapley value, which measures players' average marginal contributions in the game. Given a set of players N, by $\Pi(N)$ we mean the set of all permutations of N and for a permutation $\pi \in \Pi(N)$ we write N_i^π to mean the set of players preceding i with respect to permutation π (not including i). Shapley value of player i in game $G = (N, v)$ is defined as

$$\phi_i(G) = \frac{1}{|N|!} \sum_{\pi \in \Pi(N)} \left(v(N_i^\pi \cup \{i\}) - v(N_i^\pi) \right).$$

3 Voting Manipulation Games

We consider the scenario where in a given election $E = (C, V, \mathcal{P})$ a subset of voters $M \subseteq V$ have an established communication channel and can agree to act jointly if this can be beneficial for all of them. The two most important issues here are (1) whether the players in a group have a course of action that is more beneficial for them than truthful voting, and (2) whether the players can agree on such a course of action so that no subgroup of players can benefit by deviating from it.

To formally model this scenario, we need to define what actions are considered feasible for a coalition and how the players outside of the coalition are expected to behave.

Definition 3.1. *Given an election $E = (C, V, \mathcal{P})$, a set $M \subseteq V$ and a voting rule $\mathcal{R}$, we say that a candidate $c \in C$ is* feasible *for a coalition $S \subseteq M$ if there is a preference profile $\mathcal{P}'_S$ such that $\mathcal{R}(\mathcal{P}'_S, \mathcal{P}_{-S}) = c$. We denote the set of all candidates that are feasible for S by $F(S)$. When the voters in S vote according to a profile $\mathcal{P}'_S$ and $\mathcal{R}(\mathcal{P}'_S, \mathcal{P}_{-S}) = c$, we say that S* manipulates in favor of *c, or* supports *c.*

Note that the winner of E is feasible for any coalition $S \subseteq M$, i.e., we have $\mathcal{R}(E) \in F(S)$. Also, we emphasize that when the voters in S are trying to decide which candidates are feasible for them, they assume that all other voters (including the remaining voters in $M \setminus S$) vote truthfully. We believe that this assumption is appropriate for the following reasons. First, the issue that we are most interested in in this paper is the process of forming a manipulating coalition. We view this problem from the perspective of a voter that wants to initiate a manipulation. His primary concern is whether he can find partners who are willing to engage in a mutually beneficial collaboration with him. Once he has found such a group of like-minded voters, it is plausible that other potential manipulators—who were not invited to join the coalition—will not notice that a manipulating coalition has been formed, or will decide not to react, e.g., because, unless they coordinate among themselves, the consequences of such a reaction are uncertain. One could, of course, posit that the remaining potential manipulators will respond by forming one or more manipulating coalitions among themselves, and try to counteract the actions of the original manipulator. However, to study the resulting model, one needs to resort to non-cooperative game theory, and non-cooperative game theory models of voting appear to be hard to analyze in all but a handful of settings (see, e.g., [Desmedt and Elkind, 2010; Xia and Conitzer, 2010; Meir *et al.*, 2010]). An adversarial model, where the players in a coalition assume the worst about the actions of other players, suffers from some difficulties of its own. Thus, we decided to employ the current model because it gives a good approximation of the issues we want to focus on in this paper.

We consider the case where colluders, i.e., members of the set M in Definition 3.1, have cardinal utilities for all candidates. and can make side payments to each other. Formally, any voter $i \in M$ has a *utility function* $u_i : C \to \mathbb{R}^+ \cup \{0\}$, which satisfies $u_i(c) \ge u_i(c')$ if and only if $c \succ_i c'$. This definition can be extended to coalitions by setting $u_S(c) = \sum_{i \in S} u_i(c)$ for any $S \subseteq M$ and any $c \in C$. Note that we allow agents to assign the same utility to two different candidates. Indeed, in many voting scenarios a voter may be indifferent between some of the candidates. While the voting rule usually requires voters to provide total orders, there is no need to impose such requirements on utility functions.

Under these assumptions, a coalition $S \subseteq M$ can benefit from manipulating in favor of a candidate $c \in C$ if and only if $u_S(c) > u_S(\mathcal{R}(E))$. Indeed, if this holds, the voters in S who prefer c to $\mathcal{R}(E)$ can compensate the other voters in S by making side payments to them. Thus, a manipulating coalition should aim to elect a feasible candidate that maximizes its total utility. Formally, for any $S \subseteq M$ we set

$$\mathrm{opt}(S) = \{c \in F(S) \mid u_S(c) \ge u_S(c') \text{ for all } c' \in F(S)\}.$$

Since we have $\mathcal{R}(E) \in F(S)$ for any $S \subseteq M$, it follows that $\mathrm{opt}(S) \neq \emptyset$ for any $S \subseteq M$. In what follows, we assume that if $|\mathrm{opt}(S)| > 1$, then the manipulators in S agree on a unique alternative in S using some commonly known tie-breaking rule; therefore, abusing notation, we will treat $\mathrm{opt}(S)$ as an element of C (rather than as an element of 2^C).

We are now ready to define the (transferable utility) coalitional game that can be associated with this setting.

Definition 3.2 (Voting Manipulation Game). *Given an election $E = (C, V, \mathcal{P})$, a set $M \subseteq V$, a vector $\mathbf{u} = (u_i)_{i \in M}$ of utility functions and a voting rule $\mathcal{R}$, a* voting manipulation game *$\mathcal{R}$-$G_{E,M,\mathbf{u}}$ is a coalitional game with a set of players M and a characteristic function v given by $v(S) = u_S(\mathrm{opt}(S)) - u_S(\mathcal{R}(E))$ for any $S \subseteq M$.*

For weighted voters, the description of the game needs to be augmented with a weight vector $\mathbf{w} = (w_1, \dots, w_{|V|})$; we denote the resulting game by $\mathcal{R}$-$G_{E,M,\mathbf{u},\mathbf{w}}$.

Informally, the value of a coalition S is the maximum joint improvement over the *status quo* that the member of S can achieve, assuming other voters vote truthfully. We do not normalize the utility functions. Indeed, some voters may be indifferent to the election outcome, whereas others have strong preferences over outcomes. For computational reasons, we rescale utilities so that they are nonnegative integers.

4 Computing Coalition Values

As argued above, we always have $\mathcal{R}(E) \in F(S)$, so $v(S) \geq 0$ for any $S \subseteq M$. However, a voting manipulation game is not necessarily monotone. For example, it may be that $\mathrm{opt}(S) = \mathrm{opt}(S \cup \{i\}) = c$ for some $i \in M \setminus S$, but $\mathcal{R}(E) \succ_i c$. That is, the new voter i does not share the coalition's goal but is too insignificant to affect the action chosen by the coalition. This does not mean that i is unwilling to take part in the manipulation: the monetary transfer he gets from other manipulators induces him to participate. However, the remaining players in S may be unwilling to accept him: they can manipulate in favor of c even if i does not join, and would have to make transfers to i to keep him happy. Thus, the grand coalition M might not have a higher value than its proper subsets. Thus, we wish to identify coalitions with the highest value. A more basic question is whether we can compute the value of a given coalition. The complexity of these problems is related to the complexity of \$bribery and coalitional manipulation for the underlying voting rule.

Theorem 4.1. *Let $\mathcal{R}$ be a voting rule. There exists a poly-time algorithm for computing the characteristic function of the voting manipulation game $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ if and only if $\mathcal{R}$-*COALITIONAL MANIPULATION *is poly-time solvable.*

Proof sketch. For the "if" direction, given a coalition S, we check, for each $c \in C$, if S can make c the winner, and choose the best feasible candidate. For the "only if" direction, we set $u_i(c) = 1$, $u_i(x) = 0$ for $x \in C \setminus \{c\}$ for all $i \in M$, where c is the manipulator's preferred candidate. □

Theorem 4.2. *Let $\mathcal{R}$ be a voting rule. If $\mathcal{R}$-\$*BRIBERY *is poly-time solvable, then there exists a poly-time algorithm that given a voting manipulation game $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ computes a coalition S such that $v(S) \geq v(T)$ for any $T \subseteq M$.*

Proof sketch. Given a voting manipulation game $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ with $E = (C, V, \mathcal{P})$, for each $c \in C$ we construct an instance $I_c = (E, \mathbf{b}^c, B^c, c)$ of $\mathcal{R}$-\$BRIBERY as follows. We set $w = \mathcal{R}(E)$, $U = \max\{u_i(a) \mid i \in M, a \in C\}$. For each $i \in V$, we set $b_i^c = (n+1)U$. Further, we set $M_c = \{i \in M \mid u_i(c) > u_i(w)\}$ and for each $i \in M_c$ we set $b_i^c = 0$. Finally, for each $i \in Q_c = M \setminus M_c$ we set $b_i^c = u_i(w) - u_i(c)$, and $B^c = \sum_{i \in M_c}(u_i(c) - u_i(w))$.

If I_c is a "no"-instance of $\mathcal{R}$-\$BRIBERY, we discard this value of c. Otherwise, we use binary search to identify the smallest value $\widehat{B}^c$ such that $\widehat{I}_c = (E, \mathbf{b}^c, \widehat{B}^c, c)$ is still a "yes"-instance of $\mathcal{R}$-\$BRIBERY. Finally, we pick the candidate c that corresponds to the maximum value of $u_{M_c}(c) - u_{M_c}(w) - \widehat{B}^c$, over all non-discarded candidates, and let S be the coalition that consists of all voters in M_c together with all voters that receive non-zero bribes in $\widehat{I}_c$. Observe that we have $v(S) \geq u_{M_c}(c) - u_{M_c}(w) - \widehat{B}^c$.

Clearly, our algorithm runs in polynomial time. To see that S is a coalition with the maximum value of the characteristic function, observe that our bribery instances can be interpreted as follows: the colluders that benefit from getting c elected pool their profits from making c the winner and use them to bribe other colluders; the cheapest successful bribery corresponds to a coalition that minimizes the disutility of the colluders who prefer w to c, and therefore maximizes the total utility, among all coalitions that manipulate in favor of c. We omit the formal proof due to space constraints. □

5 Manipulating in Favor of a Given Candidate

From a candidate's perspective, a natural question is whether there exists a coalition that is willing to manipulate in her favor. One might think that the answer to this question is given by the proof of Theorem 4.2: indeed, in this proof we determine, for each candidate c, if there is a coalition that can profit from manipulating in favor of c. However, this does not necessarily answer the question above: it may happen that any coalition that *can* manipulate in favor of c would in fact prefer to manipulate in favor of some other candidate a. Indeed, it turns out that finding a coalition S such that $\mathrm{opt}(S) = c$ for a given candidate c is hard even for Plurality, and even if the number of candidates is bounded by a small constant.

Theorem 5.1. *Given a voting manipulation game* Plurality-$G_{E,M,\mathbf{u}}$ *with $E = (C, V, \mathcal{P})$ and a candidate $c \in C$, it is* NP*-complete to decide if there exists a set $S \subseteq M$ such that $\mathrm{opt}(S) = c$. The hardness result holds even if $|C| = 5$.*

Theorem 5.1's proof proceeds by a reduction from PARTITION, and uses the fact that the players' utilities are given in binary. If the number of candidates is non-constant, finding a coalition that manipulates in favor of a given candidate is hard even if all utilities are given in unary.

Theorem 5.2. *Given a voting manipulation game* Plurality-$G_{E,M,\mathbf{u}}$ *with $E = (C, V, \mathcal{P})$ and a candidate $c \in C$, deciding whether there exists a set $S \subseteq M$ such that $\mathrm{opt}(S) = c$ is strongly* NP*-complete.*

Under Plurality—as well as under many other rules—if both the number of candidates is bounded by a constant and the utilities are given in unary then finding a coalition that is willing to manipulate in favor of a particular candidate (or, determining if one exists) is easy. We postpone a formal statement of this fact till the next section, as it is closely related to the results presented there.

6 Computing Players' Power

We now explore the role of individual players in voting manipulation games. We first consider the complexity of determining whether a player is a dummy. This problem is hard even for Plurality if the number of candidates is constant, or if utilities are given in unary (but not both).

Theorem 6.1. *Given a voting manipulation game* Plurality-$G_{E,M,\mathbf{u}}$ *with* $E = (C, V, \mathcal{P})$ *and a player* $j \in M$*, it is* coNP*-complete to decide whether* j *is a dummy in* Plurality-$G_{E,M,\mathbf{u}}$. *The hardness result holds even if* $|C| = 5$.

Theorem 6.2. *Given a voting manipulation game* Plurality-$G_{E,M,\mathbf{u}}$ *with* $E = (C, V, \mathcal{P})$ *and a utility vector* $(u_i)_{i\in M}$*, and a player* $j \in M$*, it is strongly* coNP*-complete to decide whether* j *is a dummy in* Plurality-$G_{E,M,\mathbf{u}}$.

However, for a constant number of candidates we can check if a player is a dummy in pseudopolynomial time. We can also extend this result to the problem of computing a player's Shapley value (since our game is not monotone, a player may have Shapley value of 0 without being a dummy).

Theorem 6.3. *Given a voting manipulation game* $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ *with* $E = (C, V, \mathcal{P})$ *and a player* $i \in M$*, we can test if* i *is a dummy and compute* i*'s Shapley value in pseudopolynomial time as long as* $|C|$ *is bounded by a constant.*

This algorithm can be adapted to check if there exists a coalition that supports a given candidate.

Corollary 6.4. *Given a voting manipulation game* $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ *with* $E = (C, V, \mathcal{P})$ *and a candidate* $c \in C$*, we can test if there exists a coalition* S *such that* $\mathrm{opt}(S) = c$ *and if so, to compute this coalition in pseudopolynomial time as long as* $|C|$ *is bounded by a constant.*

We remark that if a player is a dummy in a voting manipulation game, it does not mean that he does not influence the outcome of the election. Indeed, by adding a player to a coalition we can change the identity of the candidate promoted by this coalition, without changing its total payoff.

Example 6.5. Suppose that our voting rule is Plurality combined with the lexicographic tie-breaking rule, $C = \{a, b, c\}$, the honest voters grant 1 point to a, 2 points to b, and 2 points to c, $M = \{1, 2\}$. Suppose that $u_1(a) = 3$, $u_1(b) = 2$, $u_1(c) = 0$ and $u_2(c) = 2$, $u_2(a) = 1$, $u_2(b) = 0$, and hence $a \succ_1 b \succ_1 c$, $c \succ_2 a \succ_2 b$. Under truthful voting, c wins. On her own, player 1 cannot change the election outcome to a, but she can change it to b, so we have $v(\{1\}) = u_1(b) - u_1(c) = 2$. On the other hand, 1 and 2 together can change the outcome to a. However, since 2 prefers c to a, he would have to be compensated. Indeed, we have $u_M(a) = 4$, $u_M(b) = 2$, $u_M(c) = 2$, so $\mathrm{opt}(\{1, 2\}) = a$ and $v(\{1, 2\}) = 2 = v(\{1\})$. Also, it is clear that $v(\{2\}) = 0$, since 2 does not want to change the election outcome. Thus, player 2 is a dummy in our voting manipulation game, yet when he joins a coalition, the coalition changes its behavior.

Conversely, a player can change the value of a coalition without changing the candidate that this coalition supports.

Example 6.6. Consider again Plurality with lexicographic tie-breaking and $C = \{a, b, c\}$. Suppose there are 10 honest voters who vote for a and 8 honest voters who vote for c, as well as four manipulators $\{1, 2, 3, 4\}$ who strictly prefer b to c to a. Set $S = \{1, 2, 3\}$. We have $\mathrm{opt}(S) = c$, $\mathrm{opt}(S \cup \{4\}) = c$, and hence $v(S \cup \{4\}) = v(S) + u_4(c) - u_4(a) > v(S)$, i.e., 4 is not a dummy. However, 4 does not have to change his vote when he joins the manipulating coalition, and neither do the voters in S. That is, 4 simply free-rides on S.

These two examples motivate the following definition.

Definition 6.7. *We say that a player* i *is* powerless *in a voting manipulation game* $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ *if for any* $S \subseteq M \setminus \{i\}$ *we have* $\mathrm{opt}(S) = \mathrm{opt}(S \cup \{i\})$.

Intuitively, a player is powerless if whenever he joins a coalition neither himself nor the players already in the coalition can benefit from changing their vote. The discussion above illustrates that a player can be a dummy without being powerless (Example 6.5) and vice versa (Example 6.6). However, checking whether a player is powerless has the same complexity as checking whether it is a dummy.

Corollary 6.8. *Given a voting manipulation game* Plurality-$G_{E,M,\mathbf{u}}$ *with* $E = (C, V, \mathcal{P})$*, it is* coNP*-complete to decide if a player is powerless. This holds even if* $|C| = 5$ *or if all utilities are given in unary. However, this problem is pseudopolynomial time-solvable if* $|C|$ *is bounded by a constant.*

7 Coalitional Stability

From game-theoretic perspective, it is important to know whether a given coalition of manipulators can be sustained, i.e., whether the core of the voting manipulation game is non-empty. This problem is easy whenever $bribery is easy.

Theorem 7.1. *If* $\mathcal{R}$-$BRIBERY *is in* P*, then there is a poly-time algorithm that given a game* $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ *with* $E = (C, V, \mathcal{P})$ *and a vector* $\mathbf{x}$ *decides whether* $\mathbf{x}$ *is in the core of* $\mathcal{R}$-$G_{E,M,\mathbf{u}}$.

Moreover, when $\mathcal{R}$-$BRIBERY is in P, we can check if the core of $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ is non-empty by formulating this problem as a linear program and using the algorithm from Theorem 7.1 as a separation oracle (see, e.g., [Elkind *et al.*, 2009] for an exposition of this technique). It is not clear if the converse of Theorem 7.1 also holds. However, we now present a construction that reduces a wide class of $\mathcal{R}$-$BRIBERY instances to testing nonmembership of an imputation $\mathbf{x}$ in the core of an $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ voting game. Whenever the bribery problem is NP-hard, our construction may be used to prove NP-hardness of testing core nonmembership.

We start with an $\mathcal{R}$-$BRIBERY instance $I = (E, \mathbf{b}, B, c)$, where $E = (C, V, \mathcal{P})$ and where we assume the following:

(a) At least one voter has bribery cost 0.

(b) There are at least two candidates and $w = \mathcal{R}(E) \neq c$.

(c) The sum of the bribery prices is greater than B.

We form a voting manipulation game $\mathcal{R}$-$G_{E,M,\mathbf{u}}$, where $M = V$ (we rename voters so that $M = \{1, \dots, n+1\}$ and so that the bribery price of voter $n + 1$ is 0). We set $\mathbf{u}$ and the imputation $\mathbf{x}$ as follows. For each voter $i \in M \setminus \{n + 1\}$ we set $u_i(w) = b_i$, $x_i = 0$, and $u_i(d) = 0$ for each candidate $d \in C \setminus \{w\}$. Also, we set $u_{n+1}(c) = B + 1$, $x_{n+1} = 0$ and

$u_{n+1}(d) = 0$ for each $d \in C \setminus \{c\}$. We see that under truthful voting $u_M(w) = \mathbf{b}(M)$ and that $v(M) = 0$ (recall that by our assumption, $\mathbf{b}(M) > B$). We also see that $\mathbf{x}$ is unstable if and only if there is a coalition S such that $c \in F(S)$ and $u_S \leq B$. Such a coalition exists if and only if our input $\mathcal{R}$-$BRIBERY instance is a "yes"-instance.

The above construction is particularly useful if $\mathcal{R}$-$BRIBERY is NP-hard, and the proof of its NP-hardness can easily be adapted to output instances that satisfy our requirements. In particular, our requirements are satisfied if the NP-hardness of $\mathcal{R}$-$BRIBERY is derived by combining Theorem 4.6 of [Faliszewski *et al.*, 2009] (a general reduction from the coalitional manipulation problem to the $bribery problem) and the fact that $\mathcal{R}$-COALITIONAL MANIPULATION is NP-hard (even if there are at least two truthful voters). Thus, we have the following corollary.

Corollary 7.2. *Suppose that* $\mathcal{R}$-COALITIONAL MANIPULATION *is* NP*-hard even if there are at least two nonmanipulators. Then given a game* $\mathcal{R}$-$G_{E,M,\mathbf{u}}$ *and an imputation* $\mathbf{x}$ *it is* NP*-hard to decide if* $\mathbf{x}$ *is not in the core of* $\mathcal{R}$-$G_{E,M,\mathbf{u}}$.

Our construction is more general and can be used, e.g., if either we do not have a complexity result for coalitional manipulation but we do have one for $bribery, or when coalitional manipulation is easy yet $bribery is NP-hard. As an example, we show that testing core nonmembership for an imputation is NP-hard for weighted Plurality.

Theorem 7.3. *Given a game Plurality-*$G_{E,M\mathbf{u},\mathbf{w}}$ *with* $E = (C, V, \mathcal{P}, \mathbf{w})$ *and a vector* $\mathbf{x}$*, it is* NP*-hard to check whether* $\mathbf{x}$ *is not in the core of Plurality-*$G_{E,M\mathbf{u},\mathbf{w}}$.

8 Conclusions

We proposed a model for collusion in voting settings that takes into account the process of forming the manipulative coalition. Our model is based on cooperative game theory and predicts which coalitions and agreements are likely to occur. Our research shows that computational problems previously studied in the context of voting manipulation, COALITIONAL MANIPULATION and $BRIBERY, which are non-game-theoretic in nature, constitute important building blocks in cooperative game-theoretic study of election manipulation.

Several questions remain open for future research. First, a key assumption of our model is that agents have comparable utilities (given in a common currency) and that they can make monetary transfers.What happens when monetary transfers are not allowed? Second, we focused on the core and the Shapley value, but other interesting solutions concepts, such as the ε-core or the nucleolus, remain to be studied. Finally, it would be interesting to examine the relation between our model and noncooperative models for voting domains, using solution concepts such as strong Nash equilibrium.

Acknowledgments

Edith Elkind was supported by NRF (Singapore) under Research Fellowship NRF RF2009-08. Piotr Faliszewski was supported by Polish Ministry of Science and Higher Education grant N-N206-378637, the Foundation for Polish Science's Homing/Powroty program, AGH University of Science and Technology grant 11.11.120.865

References

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

[Desmedt and Elkind, 2010] Y. Desmedt and E. Elkind. Equilibria of plurality voting with abstentions. In *Proceedings of EC-10*, pages 347–356, June 2010.

[Elkind *et al.*, 2009] E. Elkind, L. Goldberg, P. Goldberg, and M. Wooldridge. On the computational complexity of weighted voting games. *Annals of Mathematics and Artificial Intelligence*, 56(2):109–131, 2009.

[Faliszewski and Procaccia, 2010] P. Faliszewski and A. Procaccia. AI's war on manipulation: Are we winning? *AI Magazine*, 31(4):53–64, 2010.

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

[Faliszewski *et al.*, 2010] P. Faliszewski, E. Hemaspaandra, and H. Schnoor. Manipulation of Copeland elections. In *Proceedings of AAMAS-10*, pages 367–374, May 2010.

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

[Hazon and Elkind, 2010] N. Hazon and E. Elkind. Complexity of safe strategic voting. In *Proceedings of SAGT-10*, pages 210–221, 2010.

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

[Meir *et al.*, 2010] R. Meir, M. Polukarov, J. Rosenschein, and N. Jennings. Convergence to equilibria in plurality voting. In *Proceedings of AAAI-10*, pages 823–828, July 2010.

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

[Slinko and White, 2008] A. Slinko and S. White. Nondictatorial social choice rules are safely manipulable. In *Proceedings of COMSOC-08*, pages 403–414, 2008.

[Walsh, 2009] T. Walsh. Where are the really hard manipulation problems? The phase transition in manipulating the Veto rule. In *Proceedings of IJCAI-09*, pages 324–329, July 2009.

[Xia and Conitzer, 2010] L. Xia and V. Conitzer. Stackelberg voting games: Computational aspects and paradoxes. In *Proceedings of AAAI-10*, pages 921–926, July 2010.

[Xia *et al.*, 2009] L. Xia, V. Conitzer, A. Procaccia, and J. Rosenschein. Complexity of unweighted manipulation under some common voting rules. In *Proceedings of IJCAI-09*, pages 348–353, July 2009.

[Xia *et al.*, 2010] L. Xia, V. Conitzer, and A. Procaccia. A scheduling approach to coalitional manipulation. In *Proceedings of EC-10*, pages 275–284, June 2010.