Theoretical Computer Science ••• (••••) •••–•••

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Frugal bribery in voting

Palash Dey[a,*], Neeldhara Misra[b,*], Y. Narahari[c]

[a] Tata Institute of Fundamental Research, Mumbai, India
[b] Indian Institute of Technology, Gandhinagar, India
[c] Indian Institute of Science, Bangalore, India

ARTICLE INFO

Article history:
Received 14 June 2016
Received in revised form 24 February 2017
Accepted 28 February 2017
Available online xxxx
Communicated by L.M. Kirousis

Keywords:
Computational social choice
Voting
Bribery
Frugal
Manipulation
Algorithm
Theory

ABSTRACT

Bribery in elections is an important problem in computational social choice theory. We introduce and study two important special cases of the classical $Bribery problem, namely, Frugal-bribery and Frugal-$bribery where the briber is frugal in nature. By this, we mean that the briber is only able to influence voters who benefit from the suggestion of the briber. More formally, a voter is *vulnerable* if the outcome of the election improves according to her own preference when she accepts the suggestion of the briber. In the Frugal-bribery problem, the goal is to make a certain candidate win the election by changing *only* the vulnerable votes. In the Frugal-$bribery problem, the vulnerable votes have prices and the goal is to make a certain candidate win the election by changing only the vulnerable votes, subject to a budget constraint. We further formulate two natural variants of the Frugal-$bribery problem namely Uniform-frugal-$bribery and Nonuniform-frugal-$bribery where the prices of the vulnerable votes are, respectively, all the same or different. The Frugal-bribery problem turns out to be a special case of sophisticated $Bribery as well as Swap-bribery problems. Whereas the Frugal-$bribery problem turns out to be a special case of the $Bribery problem.

We show that the Frugal-bribery problem is polynomial time solvable for the k-approval, k-veto, and plurality with run off voting rules for unweighted elections. These results establish success in finding practically appealing as well as polynomial time solvable special cases of the sophisticated $Bribery and Swap-bribery problems. On the other hand, we show that the Frugal-bribery problem is NP-complete for the Borda voting rule and the Frugal-$bribery problem is NP-complete for most of the voting rules studied here barring the plurality and the veto voting rules for unweighted elections. Our hardness results of the Frugal-bribery and the Frugal-$bribery problems thus subsumes and strengthens the hardness results of the $Bribery problem from the literature. For the weighted elections, we show that the Frugal-bribery problem is NP-complete for all the voting rules studied here except the plurality voting rule even when the number of candidates is as low as 3 (for the STV and the plurality with run off voting rules) or 4 (for the maximin, the Copeland$^{\alpha}$ with $\alpha \in [0, 1)$, and the simplified Bucklin voting rules). In our view, the fact that the simplest Frugal-bribery problem becomes computationally intractable for many important voting rules (except the plurality voting rule) even with very few candidates is surprising as well as interesting.

* Corresponding authors.
E-mail addresses: palash@csa.iisc.ernet.in (P. Dey), mail@neeldhara.com (N. Misra), hari@csa.iisc.ernet.in (Y. Narahari).

http://dx.doi.org/10.1016/j.tcs.2017.02.031
0304-3975/

1. Introduction

In a typical voting scenario, we have a set of candidates and a set of voters reporting their *preferences or votes* which are complete rankings over the candidates. A *voting rule* is a procedure that, given a collection of votes, chooses one candidate as the winner. A set of votes over a set of candidates along with a voting rule is called an election.

Activities that try to influence voter opinions, in favor of specific candidates, are very common during the time that an election is in progress. For example, in a political election, candidates often conduct elaborate campaigns to promote themselves among a general or targeted audience. Similarly, it is not uncommon for people to protest against, or rally for, a national committee or court that is in the process of approving a particular policy. An extreme illustration of this phenomenon is *bribery* — here, the candidates may create financial incentives to sway the voters. Of course, the process of influencing voters may involve costs even without the bribery aspect; for instance, a typical political campaign or rally entails considerable expenditure.

All situations involving a systematic attempt to influence voters usually have the following aspects: an external agent, a candidate that the agent would like to be the winner, a budget constraint, a cost model for a change of vote, and knowledge of the existing election. The formal computational problem that arises from these inputs is the following: is it possible to make a distinguished candidate win the election in question by incurring a cost that is within the budget? This question, with origins in Faliszewski et al. [19–21], has been subsequently studied intensely in computational social choice literature. In particular, bribery has been studied under various cost models, for example, uniform price per vote which is known as $Bribery [19], nonuniform price per vote [18], nonuniform price per shift of the distinguished candidate per vote which is called Shift bribery, nonuniform price per swap of candidates per vote which is called Swap bribery [16]. A closely related problem known as campaigning has been studied for various vote models, for example, truncated ballots [3], soft constraints [28], CP-nets [9], combinatorial domains [27] and probabilistic lobbying [2]. The bribery problem has also been studied under voting rule uncertainty [17]. Faliszewski et al. [22] study the complexity of bribery in simplified Bucklin and Fallback voting rules. Xia [31] studies destructive bribery, where the goal of the briber is to change the winner by changing minimum number of votes. Dorn et al. [15] studies the parameterized complexity of the Swap Bribery problem and Bredereck et al. [1] explores the parameterized complexity of the Shift Bribery problem for a wide range of parameters. We recall again that the costs and the budgets involved in all the bribery problems above need not necessarily correspond to actual money traded between voters and candidates. They may correspond to any cost in general, for example, the amount of effort or time that the briber needs to spend for each voter.

1.1. Motivation

In this work, we propose an effective cost model for the bribery problem. Even the most general cost models that have been studied in the literature fix absolute costs per voter–candidate combination, with no specific consideration to the voters' opinions about the current winner and the distinguished candidate whom the briber wants to be the winner. In our proposed model, a change of vote is relatively easier to effect if the change causes an outcome that the voter would find desirable. Indeed, if the currently winning candidate is, say, a, and a voter is (truthfully) promised that by changing her vote from $c \succ d \succ a \succ b$ to $d \succ b \succ c \succ a$, the winner of the election would change from a to d, then this is a change that the voter is likely to be happy to make. While the change does not make her most favorite candidate win the election, it does improve the result from her point of view. Thus, given the circumstances (namely that of her least favorite candidate winning the election), the altered vote serves the voter better than the original one.

We believe this perspective of voter influence is an important one to study. The cost of a change of vote is proportional to the nature of the outcome that the change promises — the cost is low or nil if the change results in a better outcome with respect to the voter's original ranking, and high or infinity otherwise. A frugal agent only approaches voters of the former category, thus being able to effectively bribe with minimal or no cost. Indeed the behavior of agents in real life is often frugal. For example, consider campaigners in favor of a relatively smaller party in a political election. They may actually target only vulnerable voters due to lack of human and other resources they have at their disposal.

More formally, let c be the winner of an election and p (other than c) the candidate whom the briber wishes to make the winner of the election. Now the voters who prefer c to p will be reluctant to change their votes, and we call these votes *non-vulnerable with respect to* p — we do not allow these votes to be changed by the briber, which justifies the *frugal* nature of the briber. On the other hand, if a voter prefers p to c, then it may be very easy to convince her to change her vote if doing so makes p win the election. We name these votes *vulnerable with respect to* p. When the candidate p is clear from the context, we simply call these votes non-vulnerable and vulnerable, respectively.

The computational problem is to determine whether there is a way to make a candidate p win the election by changing *only* those votes that are vulnerable with respect to p. We call this problem Frugal-bribery. Note that there is no cost involved in the Frugal-bribery problem — the briber does not incur any cost to change the votes of the vulnerable votes. We also extend this basic model to a more general setting where each vulnerable vote has a certain nonnegative integer price which may correspond to the effort involved in approaching these voters and convincing them to change their votes. We also allow for the specification of a budget constraint, which can be used to enforce auxiliary constraints. This leads us to define the Frugal-$bribery problem, where we are required to find a subset of vulnerable votes with a total cost that is within a given budget, such that these votes can be changed in some way to make the candidate p win the election. Note