

Election Manipulation on Social Networks: Seeding, Edge Removal, Edge Addition

Matteo Castiglioni, Nicola Gatti, Giulia Landriani

Politecnico di Milano, Italy

Diodato Ferraioli

Università degli Studi di Salerno, Italy

Abstract

We focus on the *election manipulation problem* through *social influence*, where a *manipulator* exploits a *social network* to make her most preferred candidate win an election. Influence is due to information *in favor of* and/or *against* one or multiple candidates, sent by *seeds* and spreading through the network according to the independent cascade model. We provide a comprehensive study of the election control problem, investigating two forms of manipulations: *seeding* to buy influencers given a social network, and *removing* or *adding edges* in the social network given the seeds and the information sent. In particular, we study a wide range of cases distinguishing for the number of candidates or the kind of information spread over the network. Our main result is positive for democracy, and it shows that the election manipulation problem is *not* affordable in the worst-case except for trivial classes of instances, even when one accepts to approximate the margin of victory. In the case of seeding, we also show that the manipulation is hard even if the graph is a line and that a large class of algorithms, including most of the approaches recently adopted for social-influence problems, fail to compute a bounded approximation even on elementary networks, as undirected graphs with every node having a degree at most two or directed trees. In the case of edge removal or addition, our hardness results also apply to the basic case of social influence maximization/minimization. In contrast, the hardness of election manipulation holds even when the manipulator has an *unlimited* budget, being allowed to remove or add an arbitrary number of edges. Interestingly, our hardness results for seeding and edge removal/addition still hold in a *reoptimization* variant, where the manipulator already knows an optimal solution to the problem and computes a new solution once a local modification occurs, *e.g.*, the removal/addition of a single edge.

^{*}A preliminary version of this work appeared in [1], which includes some results on edge addition/removal.

1. Introduction

Nowadays, social network media are the most used, if not the unique, sources of information. This indisputable fact turned out to influence most of our daily actions, and also to have severe effects on the political life of our countries. Indeed, in many of the recent political elections around the world, there has been evidence that false or incomplete news spread through these media influenced the electoral outcome. For example, in the recent US presidential election, several studies show that, on average, 92% of Americans remembered pro-Trump false news, while 23% of them remembered the pro-Clinton fake news [2, 3]. As another example, automated accounts in Twitter spread a considerable amount of political news to alter the outcome of the 2017 French elections [4]. It also emerged that the fake news, spread over the major social media during the campaign for the 2018 Italian political election, is linked with the content of populist parties that won that election [5, 6].

The increasing use of social networks to convey inaccurate and unverified information can lead to severe and undesired consequences, as widespread panic, libelous campaigns, and conspiracies, and it represents a menace for democracy. In this scenario, some natural questions are to understand to which extent the spread of (mis)information on social network media may alter the result of a political election and how to mitigate or block it. The former problem is known in the literature as *election control through social influence*, and it has recently been the object of interest of many works in the artificial intelligence community. For instance, Sina *et al.* study a plurality voting scenario in which the voters can vote iteratively, and they show how to modify the relationship among voters to make the desired candidate to win an election [7]. Auletta *et al.* study a majority dynamics scenario and show that, in the case of two only candidates, a manipulator controlling the order in which information is disclosed to voters can lead the minority to become a majority [8, 9, 10]. Auletta *et al.* study a similar adversary in [11], showing that such a manipulator can lead a bare majority to consensus, but these results do not extend to the case with more than two candidates, as showed in [12]. Bredereck and Elkind study a majority dynamics scenario, showing how selecting seeds from which to diffuse information to manipulate a two-candidate election [13].

1.1. Main Related Works

Recently, Wilder and Vorobeychik studied a seeding problem in which all the seeds send the same information, either in favor of (*positive*) or against (*negative*) a single candidate, to make that candidate either to win or to lose, respectively, the election [14]. In particular, voters are not strategic, and ranks describe their preferences, and, given any pair of candidates c, c' such that c directly precedes c' in the rank, positive information on c' or negative information on c make c and c' switch. The diffusion of the information on the social network is described by the independent cascade model [15]. The authors provide approximation algorithms for plurality voting when the objective function is the maximization of the margin of victory. These approximation results also hold when other

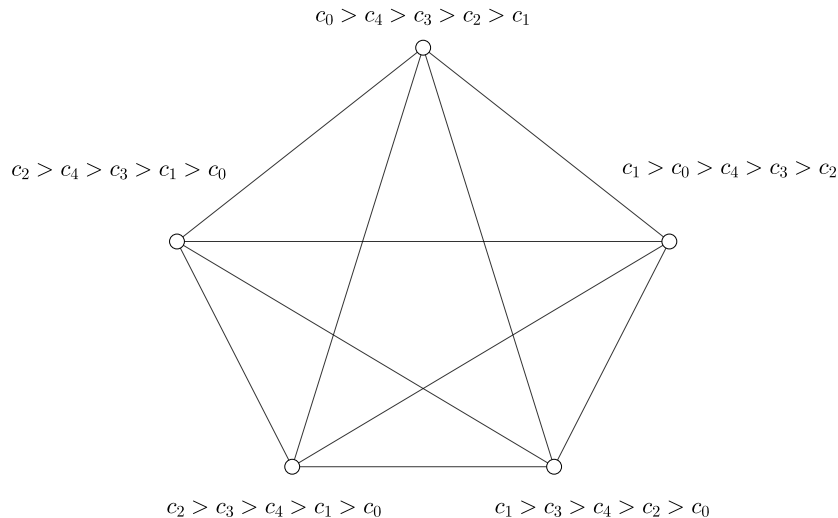


Figure 1: Clique with five voters and five candidates.

voting rules and/or other diffusion models are adopted, as showed by Corò *et al.* in [16, 17]. While the works mentioned above assume that the manipulator has complete knowledge about the problem, some recent work also deals with uncertainty on the network [18].

The works by Wilder and Vorobeychik [14] and Corò *et al.* [16, 17] present some limitations when dealing with elections with more than two candidates. A major limitation is assuming that all the seeds send the same information, and this information is on a single candidate.¹ Indeed, spreading simultaneously positive and/or negative information on multiple candidates can be, in some settings, necessary to make the manipulator’s candidate win the election, as showed in the following example.

Example 1. Consider the setting in Figure 1: there are five voters (corresponding to the nodes of the graph), five candidates c_0, c_1, \dots, c_4 , in which c_0 is the manipulator’s candidate, and each voter receives the information spread by her neighbors with probability one. A candidate gains one position in the rank thanks to positive information on her, loses one position due to negative information on her, and the manipulator has a budget sufficient for seeding two nodes, each sending information on a single candidate. When only information on a single candidate is sent, then the desired candidate c_0 cannot be made to win the election (at most, the election ends with a tie between c_0 and another candidate, both taking two votes). Instead, injecting the network with positive

¹To the best of our knowledge, the spreading of multiple information with the independent cascade model is only studied in scenarios different from election control, *e.g.*, [19].

information on c_0 and negative information on c_2 results in c_0 being the only node with two votes, and thus the winner.

Another major limitation is the assumption that seeding is the only action the manipulator can tackle to manipulate the election. However, this is not the case when the manipulator is (or collaborate with) the network media manager. In this case, the manipulator can also alter the structure of the network. In particular, she may indefinitely conceal information exchanged among two voters that are connected in the social network, or she may reveal information spread by unknown sources (*e.g.*, as sponsored content or through mechanisms as friend suggestions). That is, such a manipulator can remove or add edges in the network to obstruct or push the diffusion of information. ²

1.2. Original Contributions

In this work, we focus on the election control problem, proposing a more general model than those available in the literature and providing a comprehensive study of the complexity of manipulating the election.

1.2.1. Model and Motivation

We extend the model provided in [14], along with two different directions. First, we assume that the seeds can send different information and that the information sent by every single seed can be simultaneously positive and negative on multiple candidates. A simple interpretation is that the seeds can share different news articles and that each news article is related to a single candidate. We name the collection of information sent by every single seed *message*. As in [14], we model the diffusion of information according to a variant of the independent cascade model, capable of capturing the simultaneous spread of multiple different messages.

The second direction along which our model differs from the previous one is that voters' preferences are modeled by a scoring function that assigns, for every voter, a finite score to each candidate. Every positive (negative, respectively) information on a candidate received by a voter from each seed increases (decreases, respectively) her score. The actual score increase or decrease depends on the amount of information sent by the seed and can differ for every specific candidate. A simple interpretation is that a single news article can increase or decrease the score of a candidate by one and that every single seed can send multiple different news articles, each related to a specific candidate. As a result, differently from [14], given two candidates c, c' where c directly precedes c' , a single news article in favor of c' or against c does not necessarily make them switch as the difference of their scores can be arbitrary. On the other side, our model also allows a candidate to gain (lose, respectively) more than one position

²To the best of our knowledge, the removal or the addition of edges in the network are forms of manipulations studied only for simpler diffusion models, *e.g.*, with two candidates and simple information diffusion dynamics [13], and when no information is spread, but voters update their votes in an iterative voting process by effect of selfish voting [7, 20].

in the rank of a voter due to a large amount of received positive (negative, respectively) news articles on that candidate, sent by a single seed or by multiple seeds. Thus, a voter that is *uncertain* on c_0 can be modeled with a scoring function in which the difference between the score of the most preferred candidate and that one of c_0 is sufficiently small that the manipulator can change the voter’s preferences to make c_0 be the most preferred. Conversely, a voter that is *certain* on c_0 can be modeled with a scoring function such that the manipulator cannot (*e.g.*, due to a limitation on the news articles the manipulator can inject in the network) either make c_0 be the most preferred candidate if, initially, c_0 is not the most-preferred, or make c_0 not be the most-preferred if instead, initially, c_0 is the most preferred.

In the paper, we often refer to a special basic setting, called with *single-news-article messages*, in which all the seeds send the same information, this information is only on a single candidate and induces a score increase/decrease of one. We also use the term *unitary score distances* to refer to the case in which the difference in the score of two candidates c, c' where c directly precedes c' is exactly one. When our model is with single-news-article messages and unitary score distances and the number of candidates is two, it is directly comparable to that one studied in [14]. Instead, with three or more candidates the models are not comparable, as, differently from our model, in [14], a candidate cannot increase/decrease more than one position in the rank of a voter, even if this receives multiple positive/negative messages as sent by multiple seeds.

1.2.2. Complexity Results

We focus on the maximization of the increase in the margin of victory of the manipulator’s candidate c_0 , as done by Wilder and Vorobeychik in [14], and we provide a comprehensive study of the election manipulation problem when two forms of manipulations are possible: seeding to buy influencers given a social network, and removing or adding edges in the social network given the seeds and their messages. The manipulator is subject to budget constraints, expressing the maximum amount of cumulative news articles the seeds can spread over the network or the maximum amount of edges she can remove or add in the network. In Table 1, we summarize our main original results.

In the case of seeding, the problem is trivial when the budget available to the manipulator is unlimited, as the manipulator can make all the nodes be seeds spreading an arbitrarily large number of news articles in favor of c_0 and against all the other candidates. When instead the budget B is finite, our results depend on the scoring function of voters and the budget available to the manipulator. Initially, we observe that the setting with single-news-article messages and two candidates becomes inapproximable within a constant factor as soon as the score distances become strictly larger than one. In the general setting (*i.e.*, when messages and score distances are arbitrary), the complexity depends on the cost δ needed to make the most reluctant voter vote c_0 . We prove that whenever $\delta \leq B$, then there is a greedy poly-time algorithm guaranteeing an approximation factor ρ depending on δ . A surprisingly sharp *transition phase* occurs, instead, when $B < \delta$. In this case, no poly-time approximation

Seeding				
BUDGET	SINGLE-NEWS-ARTICLE MESSAGES		GENERAL SETTING	
	2 CANDIDATES		2 OR MORE CANDIDATES	3 OR MORE CANDIDATES
	UNITARY SCORE DISTANCES	ARBITRARY SCORE DISTANCES	$\delta \leq B, \delta$ FIXED	$B < \delta$
limited	APX [14]	\notin APX (Thm 3)	APX (Thm 4)	\notin Exp-APX (Thm 1)
Edge Removal				
BUDGET	SINGLE-NEWS-ARTICLE MESSAGES, UNITARY SCORE DISTANCES		ARBITRARY MESSAGES AND UNITARY SCORE DISTANCES	
	2 CANDIDATES		3 OR MORE CANDIDATES	2 OR MORE CANDIDATES
limited	\notin APX (Cor 2)	\notin Exp-APX (Thm 10)	\notin Exp-APX (Thm 11)	
unlimited	P (Obs 2) (†)	\notin Exp-APX (Thm 10)	\notin Exp-APX (Thm 11)	
Edge Addition				
BUDGET	SINGLE-NEWS-ARTICLE MESSAGES, UNITARY SCORE DISTANCES		ARBITRARY MESSAGES, UNITARY SCORE DISTANCES	
	2 CANDIDATES		3 OR MORE CANDIDATES	2 OR MORE CANDIDATES
limited	\notin APX (Cor 1)	\notin APX (Thm 7)	\notin Exp-APX (Thm 8)	
unlimited	P (Obs 1) (†)	\notin APX (Thm 7)	\notin Exp-APX (Thm 8)	

Table 1: Complexity results (previously known in the literature or originally provided in this paper) on the election manipulation problem through social influence. The case of seeding with unlimited budget is trivial, as discussed in the paper, and therefore omitted. Results marked with (†) also hold with arbitrary score distances.

algorithm is possible, unless $P = NP$, even when the approximation factor is a function in the size of the problem. An interpretation of such a result is that an optimal manipulation can be found in polynomial time only if every single voter is sufficiently uncertain so that her final decision on c_0 depends on the received messages. Instead, if there are voters for which c_0 cannot be made to be the most preferred candidate due to the budget constraint, then manipulation is unaffordable in polynomial time. This result poses a severe obstacle to the possibility for a manipulator to successfully alter the outcome of an election, as it is likely that there are voters who never change opinion in real-world scenarios (*e.g.*, the candidates' supporters). Even more importantly, we show that this hardness result does not hold merely for worst-case (thus, potentially, knife-edge or rare) instances. Indeed, a large class of algorithms (including most of the approaches recently adopted for social-influence problems) fail to compute an empirically bounded approximation even on elementary networks, as undirected graphs with every node having a degree at most two or directed trees. Furthermore, the hardness holds even on simple graphs, proving that maximizing the increase of margin of victory is NP-hard even on graphs as lines, and we discuss how our results extend to variants of our model.

In the case of edge removal/addition, the characterization is more intricate. We study both the case with only two candidates and the one with multiple candidates, and, in the latter case, we study both the subcase with single-news-article messages and the more realistic subcase with arbitrary messages. We show that, in any of these cases, the problem of deciding whether a set of edges to remove/add in the network exists to make the desired candidate win is hard even when the score distances are unitary. Surprisingly, these results hold even if the manipulator has an unlimited budget of edges to remove or add, except for the trivial setting in which there are two candidates and messages are single-news-

article. In this latter case, the optimal solution when the budget is unlimited is to remove all edges, if the messages are against the desired candidate, or to add all possible edges, otherwise. For the remaining cases, we formally prove that it is hard to find a set of edges to remove or add that causes an increment in the margin of victory of the desired candidate that is a constant (and, in some case, even exponential) approximation of the best possible increment that can be achieved. Our results still hold with acyclic networks.

Incidentally, in order to establish these results, we also provide new results for the basic Influence Optimization problem, that consists in maximizing or minimizing the number of nodes that receive the information spread over the network.³ We, indeed, originally prove that the minimization (maximization, respectively) variant of the problem cannot be approximated within any constant factor by removing (adding, respectively) a limited number of edges.⁴

The hardness results presented in this work are a starting point for shaping the landscape of manipulability of election through social networks. This task is fundamental to understand when and how one must design interventions to reduce the severe effects of the spread of misinformation. Although our results are positive, showing that manipulation is not affordable in the worst case, we believe that the border of manipulability can be further sharpened. We here present a seminal study along this direction, looking at manipulators that face a *reoptimization problem* [25, Chap. 4] and thus answering the question “is manipulation easier if a solution to the problem for a given instance is already available and a local modification occurs?”. Note that this is very common in the real world, in which the social relationships among voters remain essentially stable between an election and the next one. Surprisingly, we show that all our hardness results are *robust* to the knowledge of solutions in similar settings since they still hold in this reoptimization setting.

1.3. Structure of the Paper

The paper is structured as follows. Section 2 formally introduces the model and the computational problems we study. Section 3 provides our main results on the seeding problem. Section 4 provides results on edge removal, while Section 5 provides results on edge addition. Section 6 discusses the robustness of our hardness results in the case of reoptimization. Finally, Section 7 concludes the papers and describes future research directions. For the sake of presentation, some proofs are provided in Appendix A.

³In [21], the problem of adding edges to arbitrary nodes of the networks is studied, proving that the function is not submodular. In [22], two types of graph modification are investigated, adding/removing edges in order to minimize the information diffusion showing that this network structure modification problem has a supermodular objective. Heuristics for the edge removal problem have been studied in [23, 24]. However, no hardness results are known.

⁴For the sake of completeness, we mention that the Influence Optimization problem has been widely investigated when the manipulator makes seeding [15].

2. Model and Problem Statement

We have a set of candidates $C = \{c_0, c_1, \dots, c_\ell\}$ and a network of voters, represented as a weighted directed graph $G = (V, E, p)$, where V is the set of voters, E is the set of direct edges, and $p: V \times V \rightarrow [0, 1]$ denotes the strength of the potential influence among voters. In particular, for each edge (u, v) where $u, v \in V$, $p(u, v)$ returns the strength of the influence of u on v .

Each voter v assigns a *score*, by function $\pi_v : C \rightarrow \mathbb{N}$, to every candidate c_i . We assume function π_v to be injective, thus returning a different score to every candidate, formally, $\pi_v(i) \neq \pi_v(j), \forall c_i, c_j \in C$. The score $\pi_v(i)$ models how much voter v likes candidate c_i and induces, for voter v , a strict preference ordering over the candidates. Thus, $\pi_v(i) > \pi_v(j)$ models that voter v (strictly) prefers c_i to c_j . Sometimes we will denote with $\langle \pi_v(0), \dots, \pi_v(\ell) \rangle$ the score vector of voter v . We will say that the scores have *unitary score distances* if $\pi_v(i) \in \{0, \dots, \ell - 1\}$ for every voter v and for every candidate c_i .

The election is based on *plurality voting*, where every voter casts a single vote for a single candidate, and the candidate that received the largest number of votes wins the election. We assume voters to be *myopic*, casting a vote for the candidate with highest score in their preference ordering. For each candidate $c \in C$, we denote with V_c the set of voters that rank c as first, formally, $V_c = \{v \in V \mid c = \operatorname{argmax}_{c_i \in C} \pi_v(i)\}$.

Let $S \subseteq V$ be a subset of voters said *seeds*. Every seed s can be selected to initiate the diffusion of information about multiple candidates. We denote with $m_s = (q_0, \dots, q_\ell)$ the *message* of $s \in S$, where $q_i \in \mathbb{Z}$, with $q_i > 0$ ($q_i < 0$, respectively) representing that s initiates the diffusion of q_i *positive* (*negative*, respectively) news articles on c_i , and $q_i = 0$ representing that s does not send any information about c_i . Positive information on a candidate increases the scores that voters assign to that candidate by $|q_i|$, while negative information does the reverse. We use $m_s(i) \in \mathbb{Z}$ to denote information sent by s on candidate c_i and $M = \cup_{s \in S} m_s$ to denote the whole information sent by seeds. We denote with $|m_s| = \sum_{c_i \in C} |m_s(i)|$ the number of news articles sent by s . Similarly, $|M| = \sum_{s \in S} |m_s|$. If $m_s = m_{s'}$ for every pair of seeds $s, s' \in S$, and, for all the candidates c_i except c_j , it holds $m_s(i) = 0$, while for c_j it holds $m_s(j) \in \{+1, -1\}$, then we say that we are in the setting with *single-news-article messages*.

2.1. Diffusion Model

Given a pair of seeds/messages (S, M) , messages are supposed to spread over the network according to a *multi-issue independent cascade* (MI-IC) model. Roughly speaking, in this model, each seed s propagates the message m_s to her neighbors. Then, a voter v with $v \notin S$, receiving a message from s , accepts the information that this message carries with probability $p(s, v)$. If voter v accepts the message, we say that v is *activated* by s . In her turn, each just activated voter v sends the received messages to her neighbors u that can activate with

probability $p(v, u)$ if not activated in the past and, then, voter v becomes *inactive*. The process continues as long as there is some active voter, and it is repeated for every message m_s sent by one of the seeds.

Formally, given graph $G = (V, E, p)$, we define the *live-graph* $H = (V, E')$, where each edge $(u, v) \in E$ is included in H with probability $p(u, v)$. Moreover, for every $s \in S$, we introduce a set $A_{m_s}^t \subseteq V$ composed of the *active* voters at time t due to message m_s . Every set $A_{m_s}^t$ is initialized with the seed sending the corresponding message for $t = 0$, *i.e.*, $A_{m_s}^0 = \{s\}$, and the empty set for $t > 0$. At every time $t \geq 1$, set $A_{m_s}^t$ is defined as follows: for every edge $(u, v) \in E'$, we consider the set $\mathcal{M}_{(u,v)} \subseteq M$ of messages m_s such that $u \in A_{m_s}^{t-1}$ —and thus u has just been activated by m_s —and $v \notin \bigcup_{i < t} A_{m_s}^i$ —and thus v has never been activated by m_s ; then for each (u, v) such that $\mathcal{M}(u, v)$ is not empty, we add v to $A_{m_s}^t$ for every $m_s \in \mathcal{M}(u, v)$. The diffusion process of message m_s terminates at time T_{m_s} when $A_{m_s}^{T_{m_s}} = \emptyset$. Finally, the cascade terminates when the diffusion of every message m_s terminates. A voter that activates at some t is said *influenced*. Note that, when the messages are single-news-article, there are two candidates, and the score distances are unitary, this process reduces to the renowned independent cascade model [15].

2.2. Preference Revision

When a voter v accepts a message received by a neighbor, her preferences can change. Let us now denote with $R \subseteq M$ a set of received messages. A *ranking revision function* ϕ associates each pair (π, R) with a new ranking π' obtained by revising ranking π according to the set of received messages R . We use a *score-based* ranking revision function in which a positive (negative) message m_s on a candidate c_i increases (decreases) her score by $m_s(i)$. Formally, each voter updates every candidate's score as follows:⁵

$$\pi_v(i) \leftarrow \pi_v(i) + \sum_{m_s \in R} m_s(i).$$

According to our assumption on π_v , we require that, at the end of the diffusion process, no pair of candidates c_i, c_j has the same value of π_v . We can obtain such a property, *e.g.*, by breaking ties according to some rule and slightly tilting scores so that they satisfy the tie-break outcome. For the sake of simplicity, we brake ties in favor of the candidate ranked last before the diffusion process. Such a tie-breaking rule can be obtained by slightly perturbing the initial score with a multiplicative factor $(1 - \epsilon)$, where ϵ is a sufficiently small positive constant, *e.g.*, $\epsilon = \frac{1}{1 + \max_{v,i} \pi_v(i)}$, and then apply the update rule as

$$\pi_v(i) \leftarrow (1 - \epsilon) \pi_v(i) + \sum_{m_s \in R} m_s(i).$$

⁵It is easy to see that every hardness result related to this model keeps to hold even when we allow the same message causes a different score increment (decrement) to different voters, or if received by different neighbors, or if sent by different seeds.

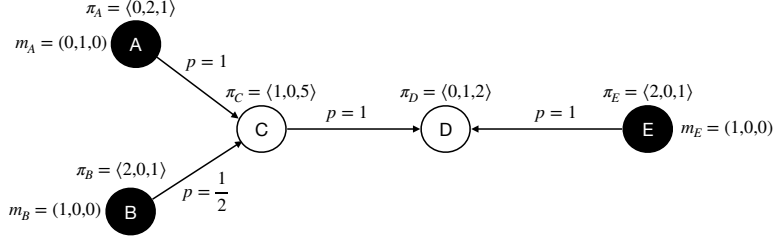


Figure 2: Example of an election with three candidates c_0, c_1, c_2 . Black nodes represent seeds: node A sends a positive message on c_1 , while nodes B and E send a positive message on c_0 . The tuples $\langle \pi_v(0), \pi_v(1), \pi_v(2) \rangle$ above the nodes are the voters' preferences.

Given a seed set S , a set M of messages, a set E of edges, and a live graph H , $\pi_v^*(i, S, M, E, H)$ denotes for every voter $v \in V$, the score of candidate c_i at the end of the MI-IC diffusion (*i.e.*, after the preference revision). Moreover, for each candidate $c \in C$, we denote with V_c^* the set of voters for which c is ranked as first after the preference revision, *i.e.*, $V_c^*(S, M, E, H) = \{v \mid \operatorname{argmax}_{c_i} \pi_v^*(i, S, M, E, H) = c\}$. Finally, we define the *margin of victory* MoV of (S, M, E, H) as

$$\operatorname{MoV}(S, M, E, H) = \left| V_{c_0}^*(S, M, E, H) \right| - \max_{c \neq c_0} \left| V_c^*(S, M, E, H) \right|.$$

Given a live-graph H , MoV returns the number of votes that c_0 needs to win the election, if the first term is smaller than the second, and the advantage of c_0 with respect to the second-best ranked candidate, otherwise.

Next we provide an example of the concept defined above.

Example 2. Consider Figure 2, depicting the connections among five voters.

Two different live-graphs H_1 and H_2 are possible depending on whether or not B influences C. This happens with probability $\frac{1}{2}$.

In H_1 , B does not influence C and C receives only a positive news article on c_1 , thus increasing the score of c_1 by 1. However, C has a very high evaluation of candidate c_2 and keeps to prefer c_2 over c_0 and c_1 . Instead, D updates her score to $\langle 1, 2 - \epsilon, 2 - 2\epsilon \rangle$ and votes for c_1 . Thus, at election time, c_0 has 2 votes (B and E), c_1 has 2 votes (A and D) and c_2 has one votes (C), and therefore $\operatorname{MoV}(S, M, E, H_1) = 2 - \max\{2, 1\} = 0$.

In H_2 , B influences C and C receives a positive news article on c_0 and a positive news article on c_1 . However, C keeps to prefer c_2 over c_0 and c_1 . Voter D receives a positive news article on c_1 and two positive news articles on c_0 , thus updating the scores to $\langle 2, 2 - \epsilon, 2 - 2\epsilon \rangle$ and then voting for c_0 . Hence, $\operatorname{MoV}(S, M, E, H_2) = 3 - \max\{1, 1\} = 2$.

2.3. Election Control Problem

The election control problem involves a single agent (*i.e.*, the manipulator) whose objective is to spend a *budget* B to make c_0 win the election. We consider

two different manipulation strategies: seeding and network modification by edge removal/addition. For the sake of simplicity, we assume that the cost incurred by the manipulator for seeding is one for every single news article sent by each seed and therefore the cumulative cost for seeding is $|M|$, while the cost for network modification is equal to the number of removed/added edges. We study each form of manipulation singularly. Nevertheless, it is easy to see that the results also extend to the case with multiple simultaneous forms of manipulation.

We next formally state the problems we study in the paper.

Definition 1 (ELECTION-CONTROL-BY-SEEDING (ECS)). *Given an election scenario $(C, G, \{\pi_v\})$ and budget $B \in \mathbb{N}$, the goal is finding a set S of seeds and messages m_s for every $s \in S$, with $|M| \leq B$, to maximize $\mathbb{E}_H[\Delta_{\text{MoV}}^S(S, M, H)]$, where $\Delta_{\text{MoV}}^S(S, M, H) = \text{MoV}(S, M, E, H) - \text{MoV}(\emptyset, \emptyset, E, H)$ is the increase of MoV due to the messages send by seeds S .*

Definition 2 (ELECTION-CONTROL-BY-EDGE-REMOVAL (ECER)). *Given an election scenario $(C, G, \{\pi_v\}, S, M)$ and budget $B \in \mathbb{N} \cup \{\infty\}$, the goal is finding $E' \subseteq E$ with $|E'| \leq B$ to remove from graph G to maximize $\mathbb{E}_H[\Delta_{\text{MoV}}^-(E', H)]$, where $\Delta_{\text{MoV}}^-(E', H) = \text{MoV}(S, M, E \setminus E', H) - \text{MoV}(S, M, E, H)$ is the increase of MoV due to the removal of edges E' .*

Definition 3 (ELECTION-CONTROL-BY-EDGE-ADDITION (ECEA)). *Given an election scenario $(C, G, \{\pi_v\}, S, M)$ and budget $B \in \mathbb{N} \cup \{\infty\}$, the goal is finding E' with $E' \cap E = \emptyset$ and $|E'| \leq B$ to add to G to maximize $\mathbb{E}_H[\Delta_{\text{MoV}}^+(E', H)]$, where $\Delta_{\text{MoV}}^+(E', H) = \text{MoV}(S, M, E \cup E', H) - \text{MoV}(S, M, E, H)$ is the increase of MoV due to the addition of edges E' .*

An algorithm A is said to always return a ρ -approximation for an ECS problem with $\rho \in [0, 1]$ potentially depending on the size of the problem, if, for each instance of the problem, it returns a feasible (S, M) such that $\mathbb{E}_H[\Delta_{\text{MoV}}(S, M, H)] \geq \rho \mathbb{E}_H[\Delta_{\text{MoV}}(S^*, M^*, H)]$. A similar definition holds for all the other optimization problems.

2.4. Influence Optimization

Incidentally, our analysis of the ECER and ECEA problems allow us to provide results also on the (more general) influence maximization/minimization problems when the manipulator can either *remove* or *add edges*. To formally describe these problems, we need to define function $\chi : S \times E \times H \rightarrow \mathbb{R}_+$ returning the number of influenced nodes with seeds S , edges E and live graph H . When the set of edges E is fixed (e.g., in seeding), we will use $\chi(S, H) = \chi(S, E, H)$, removing the dependence from E . Finally, with abuse of notation, we also define $\chi(S, E) = \mathbb{E}_H[\chi(S, E, H)]$. We have the following two problems.

Definition 4 (INFLUENCE-MINIMIZATION-BY-EDGE-REMOVAL (IMER)). *Given a setting (G, S, M) and budget $B \in \mathbb{N} \cup \{\infty\}$, the goal is finding a set $E' \subseteq E$ with $|E'| \leq B$ to remove from graph G to maximize $\Delta I^-(E') = \chi(S, E) - \chi(S, E \setminus E')$.*

Definition 5 (INFLUENCE-MAXIMIZATION-BY-EDGE-ADDITION (IMEA)). *Given a setting (G, S, M) and budget $B \in \mathbb{N} \cup \{\infty\}$, the goal is finding a set E' with $E' \cap E = \emptyset$ and $|E'| \leq B$ to add to graph G to maximize $\Delta I^+(E') = \chi(S, E \cup E') - \chi(S, E)$.*

3. Seeding Complexity

We characterize the computational complexity of the ECS problem. Unless specified otherwise, the results provided in this section refer to the general setting when both messages and score distances are arbitrary. Our characterization is based on the parameter $\delta = \max_{v \in V, c_i \neq c_0} \{\pi_v(i) - \pi_v(0)\}$, representing the cost the manipulator needs to spend to convince the most reluctant voter to vote for c_0 . We introduce the following definition that is useful to describe the hardest instances of the ECS problem.

Definition 6. *An ECS problem instance is said hard to manipulate if $B < \delta$.*

3.1. Inapproximability Results

We introduce the SET-COVER problem, that is well known to be NP-hard, to prove the hardness of ECS.

Definition 7 (SET-COVER). *Given a set $N = \{z_1, \dots, z_n\}$ of n elements, a collection $X = x_1, \dots, x_g$ of sets with $x_i \subset N$, and a positive integer h , the objective is to select a collection $X^* \subset X$, $|X^*| \leq h$ with $\cup_{x_i \in X^*} x_i = N$.*

Theorem 1. *Given the set of ECS instances said hard to manipulate and with at least three candidates, for any $\rho > 0$ even depending on the size of the problem, there is not any poly-time algorithm returning a ρ -approximation to the ECS problem, unless $P = NP$.*

Proof. The proof uses a reduction from SET-COVER. Given an instance of SET-COVER, we build an instance of the election control problem with 3 candidates as follows.⁶ The voters' network G , showed in Figure 3, consists of three disconnected components, that we denote as G_1 , G_2 , and G_3 . Note that all edges of G have $p(u, v) = 1$.

We set the budget $B = h + 1$. The component G_1 has $g + n$ nodes and it is used to model the SET-COVER instance. Indeed, for each $z_i \in N$, we have in G_1 a node v_{z_i} ; moreover, for each $x_i \in X$, we have in G_1 a node v_{x_i} with an edge toward v_z for each $z \in x_i$. The preferences of all voters v corresponding to nodes in G_1 are: $\pi_v(0) = 0$, $\pi_v(1) = B$, $\pi_v(2) = B + 1$.

The component G_2 is a clique of $n + h + 1$ nodes with preferences: $\pi_v = \langle 0, B + 1, B \rangle$. The component G_3 is a clique of $n + 2g - h + 3$ nodes, such that

⁶If $|C| > 3$, we set $\pi_v(i) = \max\{\pi_v(0), \pi_v(1), \pi_v(2)\} - B - 1, \forall i \in \{3, \dots, C - 1\}, v \in V$.

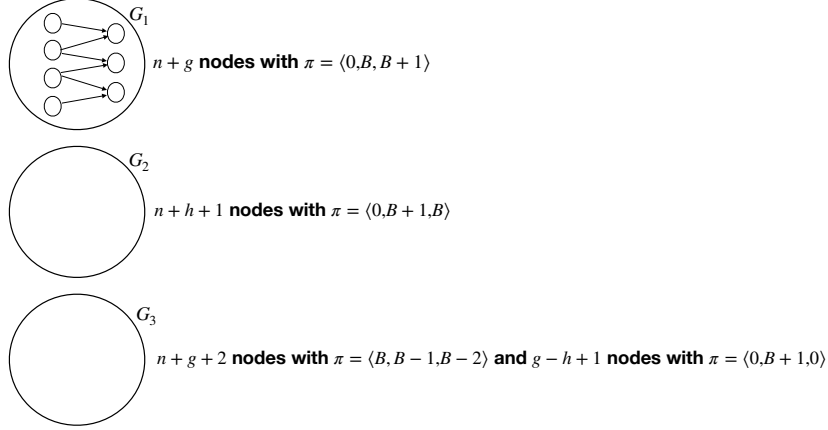


Figure 3: Structure of the election control problem used in the proof of Theorem 1.

$n + g + 2$ nodes have a preference ranking $\pi_v = \langle B, B - 1, B - 2 \rangle$ and $g - h + 1$ nodes have a preference ranking $\pi_v = \langle 0, B + 1, 0 \rangle$.

Note that $|V_{c_0}| = g + n + 2$, $|V_{c_1}| = g + n + 2$, and $|V_{c_2}| = g + n$. Hence, $\text{MoV}(\emptyset, (), E, H) = 0$.

We next prove that this instance allows a feasible solution (S^*, M^*) with $\text{MoV}(S^*, M^*, E, H) > 0$ if and only if there is a solution of the SET-COVER instance of size at most h .

(If) Let $X^* \subseteq X$ be the solution of SET-COVER of size h (i.e., $|X^*| = h$ and $\cup_{x_i \in X^*} = N$).⁷ Then we set (S^*, M^*) as follows: for every $x_i \in X^*$, we include $v_{x_i} \in S^*$ and we set $m_{v_{x_i}}^*$ such that $q_0 = 0$, $q_1 = 1$, and $q_2 = 0$; moreover, we include in S^* an arbitrary node $v \in G_2$ and we set m_v^* such that $q_0 = 0$, $q_1 = 0$, and $q_2 = 1$.

From the above arguments, it directly follows that (S^*, M^*) is feasible. We next show that $\text{MoV}(S^*, M^*, E, H) > 0$. Indeed, the diffusion of messages leads each voter corresponding to nodes in G_2 to prefer c_2 to c_1 . Moreover, the dynamics leads $h + n$ voters in G_1 (i.e., the seeds and the ones corresponding to elements $z_i \in N$) to prefer c_1 to c_2 . Hence, $|V_{c_1}^*(S^*, M^*, H)| = |V_{c_1}| - |G_2| + h + n = g + n + 2 - n - h - 1 + n + h = g + n + 1$, and $|V_{c_2}^*(S^*, M^*, H)| = |V_{c_2}| + |G_3| - h - n = g + n + 1$. Hence, $\text{MoV}(S^*, M^*, E, H) = 1$, as desired.

(Only if) Suppose that there exists a pair (S^*, M^*) such that it holds $\text{MoV}(S^*, M^*, E, H) > 0$. Note that, since $\delta > B$, c_0 cannot gain votes and $|V_{c_0}^*(S^*, M^*, H)| = |V_{c_0}|$. Hence, in order to have $\text{MoV}(S^*, M^*, E, H) > 0$, it must be the case that the number of voters whose most-preferred candidate is c_1 decreases by at least one unit and the number of voters whose most-preferred

⁷If there is a solution of SET-COVER X^* of size less h , then we can achieve a solution of SET-COVER of size exactly h , by padding X^* with arbitrary element in $X \setminus X^*$.

candidate is c_2 increases by at most one unit.

Since, c_2 has to take at least one vote and she cannot take votes in G_3 , c_2 must take voters in G_2 . Since G_2 is a clique, it must be that all votes of c_1 are taken by candidate c_2 .

Thus, c_1 loses all its voters in G_2 in favor of c_2 . Note that a single message is sufficient (a positive message for c_2) to this aim. However, this implies that c_2 must lose $n + h$ voters in G_1 , otherwise $|V_{c_2}^*(S^*, M^*, E, H)| > g + n + n + h + 1 - (n + h)$ and thus $\text{MoV}(S^*, M^*, E, H) \leq 0$, that contradicts our hypothesis. Observe that these votes must be necessarily lost in favor of c_1 .

Hence, we are left with h available messages to make $n + h$ voters to change their vote from c_2 to c_1 . Observe that, in order to make a voter to change, it is sufficient a single message (a positive message for c_1). However, if less than h seeds sending this message are located among nodes v_{x_i} for $x_i \in X$, then less than $n + h$ voters will change their mind (since nodes v_{x_i} for $x_i \in X$ have no in-going edges).

Finally, we must have that the h seeds in G_1 are neighbors of every node v_{z_i} for $z_i \in N$. Hence, the set $X^* = \{x_i : v_{x_i} \in S^*\}$ has size h and, by construction of G_1 , $\bigcup_{x \in X^*} x = N$, *i.e.*, X^* is a solution of SET-COVER of size at most h .

Hence, we can conclude that a feasible solution (S^*, M^*) satisfying the property $\Delta_{\text{MoV}}(S^*, M^*, H) > 0$ exists if and only if a solution for the SET-COVER instance exists. Note also that if a solution with $\Delta_{\text{MoV}}(S^*, M^*, H) > 0$ exists, then $\Delta_{\text{MoV}}(S, M, H) > 0$ even for any ρ -approximate solution (S, M) , regardless of the value of ρ . Thus, if a polynomial time ρ -approximate algorithm for election control problem exists, then the SET-COVER problem can also be solved in poly-time, implying that $\text{P} = \text{NP}$. \square

Theorem 1 essentially states that there is no chance that a manipulator designs an algorithm allowing her to maximize the increment in the margin of victory of the desired candidate in the set of instances in which there is no way for making c_0 become the most preferred of any voter. However, Theorem 1 does not rule out that the worst-case instances are very rare and/or knife-edge. However, we show that simply algorithms will fail even on very simple instances. Specifically, we show that if the manipulator greedily chooses the messages to send, then her approach fails even for simple graphs, namely graphs with all nodes having a degree two or trees.

In details, given a set S of seeds and corresponding messages M , we denote as $\mathcal{F}(S, M)$ the set of pairs (s, m_s) , with $s \notin S$ such that either

$$\mathbb{E}_H [\text{MoV}(S \cup \{s\}, (M, m_s), E, H)] > \mathbb{E}_H [\text{MoV}(S, M, E, H)]$$

or

$$\mathbb{E}_H [V_{c_0}^*(S \cup \{s\}, (M, m_s), E, H)] > \mathbb{E}_H [V_{c_0}^*(S, M, E, H)].$$

That is, $\mathcal{F}(S, M)$ includes all the ways of augmenting a current solution so that either the margin of victory of c_0 or the number of her votes increases. Then, we say that an algorithm for the election control problem uses the *greedy* approach, if it works as follows: it starts with $S = \emptyset$ and $M = ()$; until the set $\mathcal{F}(S, M)$ is

not empty, choose one $(s, m_s) \in \mathcal{F}(S, M)$ and set $S = S \cup \{s\}$ and $M = (M, m_s)$. We show that every algorithm in this class fails even for elementary networks (the proof of the following proposition is provided in Appendix A).

Proposition 1. *For any $\rho > 0$ even depending on the size of the problem, no algorithm following the greedy approach returns a ρ -approximation to the ECS problem, even in undirected graphs in which each node has degree at most 2.*

A similar result holds even by considering directed trees (the proof of the following proposition is provided in Appendix A).

Proposition 2. *For any $\rho > \frac{38}{|V|}$, no algorithm following the greedy approach returns a ρ -approximation to the ECS problem, even in directed trees.*

We recall that the greedy algorithms are essentially the only known algorithms guaranteeing bounded approximations for many problems related to the election control problem, such as the well-known influence maximization problem [15]. Hence, even if an algorithm exists enabling the manipulator to control the election in many instances, the propositions above show that new approaches are necessary to design it.

Now, we provide further evidence of the hardness of the problem even in simple graphs, showing that maximizing the expected Δ_{MOV}^S is NP-hard even on a line. We recall that, while Influence Maximization by Seeding is NP-hard with arbitrary graphs, there exists a polynomial-time algorithm when the graph is a line [26]. The proof of the following theorem is provided in Appendix A.

Theorem 2. *The ECS problem with at least four candidates is NP-hard even on line graphs.*

We conclude the section proving that when we restrict to single-news-article-messages the ECS problem is hard even in the simple model in which there are two candidates and the score distances are two. We reduce from DENSEST-K-SUBGRAPH whose definition follows.

Definition 8 (DENSEST-K-SUBGRAPH (DKS)). *Given an indirect graph $G = (X, N)$, find the set X^* of k vertexes that maximizes $d(X^*) = |E'|$, where $G(X^*) = (X^*, E')$ is the subgraph of G with vertexes X^* .*

The proof of the following theorem is provided in Appendix A.

Theorem 3. *If there is a $\rho > 0$ approximation algorithm for the ECS problem with single-news-article messages, two candidates, and arbitrary scores, then there is a ρ -approximation algorithm for DKS.*

Manurangsi shows that there is no constant-factor polytime approximation algorithm for the DKS problem unless the Exponential Time Hypothesis is false [27]. Therefore, it is unlikely that the specific ECS problem considered above is approximable within a constant factor. This is in stark contrast with the known constant approximation algorithm existing in the setting when we further constrain score distances to be unitary [14].

3.2. Approximation Results

We next show that the condition we used in the previous section to characterize hard instances is tight. Indeed, by dropping that condition, we can design poly-time approximation algorithms. Moreover, these algorithms turn out to follow the greedy approach that we proved to fail even for simple structures in hard-to-manipulate instances.

Theorem 4. *Let $\delta \leq B$. There is a greedy poly-time algorithm returning a ρ -approximation to the ECS problem, with*

$$\rho = \frac{B - \delta + 1}{2\delta B} \left(1 - \frac{1}{e}\right).$$

Proof. Let m^* with $|m^*| = \delta$ be the message that cause each voter to vote for c_0 , whatever was the ranking before the reception of this message. Our algorithm selects $\lfloor \frac{B}{\delta} \rfloor$ seeds through the greedy algorithm to maximize $\mathbb{E}_H[\chi(S, H)]$ (i.e., take at each time the seed that most increases this quantity), and let each of them to send the message m^* . It directly follows that this algorithm runs in poly-time in greedy fashion.

In order to formally prove the approximation factor of this algorithm for the election control problem, let us denote with \hat{S} the set of seeds returned by our algorithm, with S^* the set of seeds maximizing $\mathbb{E}_H[\Delta_{\text{MoV}}(S, M^*, H)]$, with S' the set of seeds of size B that maximizes $\mathbb{E}_H[\chi(S, H)]$ and with S'' the set of seeds of size $k = \lfloor \frac{B}{\delta} \rfloor$ that maximizes $\mathbb{E}_H[\chi(S, H)]$.

It is known that the function $\mathbb{E}_H[\chi(S, H)]$ is monotone and submodular on S [15], i.e., $\mathbb{E}_H[\chi(S, H)] \leq \mathbb{E}_H[\chi(T, H)]$ and $\mathbb{E}_H[\chi(S \cup \{x\}, H)] - \mathbb{E}_H[\chi(S, H)] \geq \mathbb{E}_H[\chi(T \cup \{x\}, H)] - \mathbb{E}_H[\chi(T, H)]$ for every $S \subseteq T$ and every $x \notin T$. Consequently, the greedy algorithm is known to return, for every k , a set of k seeds whose influence is an $(1 - \frac{1}{e})$ -approximation of the maximum expected influence achievable with k seeds [15]. Hence, we have that:

$$\mathbb{E}_H[\chi(\hat{S}, H)] \geq \left(1 - \frac{1}{e}\right) \mathbb{E}_H[\chi(S'', H)]. \quad (1)$$

Note that $|V_c| - \mathbb{E}_H[|V_c^*(S^*, M^*, H)|] \leq \mathbb{E}_H[\chi(S^*, H)]$ for every $c \neq c_0$, since at most one vote can be lost by c for every influenced node in graph H . Then we have that

$$\begin{aligned} & \max_{c \neq c_0} |V_c| - \mathbb{E}_H \left[\max_{c \neq c_0} |V_c^*(S^*, M^*, H)| \right] \\ & \leq \max_{c \neq c_0} \left\{ |V_c| - \mathbb{E}_H[|V_c^*(S^*, M^*, H)|] \right\} \\ & \leq \mathbb{E}_H[\chi(S^*, H)]. \end{aligned} \quad (2)$$

A similar argument proves that

$$\mathbb{E}_H[|V_{c_0}^*(S^*, M^*, H)|] - |V_{c_0}| \leq \mathbb{E}_H[\chi(S^*, H)]. \quad (3)$$

Moreover, by submodularity of χ , it holds that $\frac{\mathbb{E}_H[\chi(S', H)]}{|S'|} \leq \frac{\mathbb{E}_H[\chi(S'', H)]}{|S''|}$. Hence, since $|S'| = B$ and $|S''| = \lfloor \frac{B}{\delta} \rfloor \geq \frac{B - \delta + 1}{\delta}$, we achieve that

$$\mathbb{E}_H[\chi(S', H)] \leq \frac{\delta B}{B - \delta + 1} \mathbb{E}_H[\chi(S'', H)]. \quad (4)$$

Moreover, by definition of Δ_{MoV}^S , we have

$$\begin{aligned} \mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)] &= \\ \mathbb{E}_H[|V_{c_0}^*(S^*, M^*, H)|] &- \mathbb{E}_H\left[\max_{c \neq c_0} |V_c^*(S^*, M^*, H)|\right] - \left(|V_{c_0}| - \max_{c \neq c_0} |V_c|\right). \end{aligned}$$

Hence, we directly achieve that

$$\begin{aligned} \mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)] &= \\ (\mathbb{E}_H[|V_{c_0}^*(S^*, M^*, H)|] &- |V_{c_0}|) + \left(\max_{c \neq c_0} |V_c| - \mathbb{E}_H[\max_{c \neq c_0} |V_c^*(S^*, M^*, H)|]\right). \end{aligned}$$

Then, have that

$$\begin{aligned} \mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)] &\leq 2 \mathbb{E}_H[\chi(S^*, H)] && \text{(by (2) and (3))} \\ &\leq 2 \mathbb{E}_H[\chi(S', H)] && \text{(by definition of } S') \\ &\leq \frac{2 \delta B}{B - \delta + 1} \mathbb{E}_H[\chi(S'', H)] && \text{(by (4))} \\ &\leq \frac{2 \delta B}{B - \delta + 1} \left(1 - \frac{1}{e}\right)^{-1} \mathbb{E}_H[\chi(\hat{S}, H)] && \text{(by (1))} \\ &\leq \frac{2 \delta B}{B - \delta + 1} \left(1 - \frac{1}{e}\right)^{-1} \mathbb{E}_H[\Delta_{\text{MoV}}^S(\hat{S}, M^*, H)], \end{aligned}$$

where the last inequality follows from the fact that, by definition of m^* , all the influenced nodes will vote for c_0 . \square

Notice that Theorem 4 guarantees a constant approximation factor whenever δ is fixed. Our algorithm heavily depends on the possibility that the seeds can send messages with information on multiple candidates. Indeed, Theorem 3 shows that, without this possibility, it is unlikely that the ECS problem is approximable within a constant factor even with two candidates.

3.3. Seeding Complexity in Variants of the Model

We describe some extensions and variants of our model and show how most of the results presented in the previous section extend to these settings.

3.3.1. *Bribed Seeds*

In our model, seeds act as initiators of positive and/or negative messages about the candidates. However, apart from that, their behavior is exactly the same as any other node in the network. In particular, the messages that they receive affect their preference ranks and, consequently, their vote. We also study a variant, in which seeds are *bribed* (i.e., for each seed, their preferred candidate is set by the manipulator, and it is independent from her initial preference rank, and from the messages that she sends and receives).

It directly follows that the reduction described in the proof of Theorem 1 does not work in this variant. However, we next show that, even in this variant, the ECS problem is inapproximable.

Theorem 5. *Given the set of ECS instances said hard to manipulate and with at least three candidates, for any $\rho > 0$, there is not any poly-time algorithm returning a ρ -approximation to the ECS problem with bribed seeds, unless $P = NP$.*

Proof. Consider the reduction described in the proof of Theorem 1, except that now each node is enlarged into a clique of size $(h+1)\rho'$, where $\rho' > \rho$. Hence, if a set cover of size at most h exists, then, $\Delta_{\text{MOV}}(S^*, I^*, H) \geq (h+1)\rho'$, otherwise the only nodes that eventually change opinion are the seeds, that are at most $h+1$. Thus any ρ -approximate algorithm must be able to distinguish these two cases and thus solves the SET-COVER problem in polynomial time. \square

Instead, it is easy to check that Theorem 4 is unaffected by the fact that seeds are bribed, and thus a constant approximation is still possible when $\delta \leq B$.

3.3.2. *Other Objective Functions*

In addition to the maximization of the increase in the margin of victory, alternative objective functions, previously studied in [14], may be of interest.

For example, one may want to maximize the increase in the probability of victory. For this objective function, it is not trivial to see that Theorem 1 keeps holding. However, notice that this objective function makes the problem even harder than maximizing the increase in the margin of victory. Indeed, for the latter objective, Theorem 4 implies that a $\frac{1}{2}(1 - \frac{1}{e})$ -approximation can be computed in poly-time when only two candidates are involved. It is instead not hard to see that, to maximize the increase in the probability of victory with only two candidates, it is sufficient that all selected seeds send the same message. Hence, for two candidates, maximizing the increase in the probability of victory in our setting is the same as doing it in the setting studied in [14]. Hence, the problem cannot be approximated, unless $P = NP$, within a factor $\rho > 0$, even for two only candidates.

An apparently weaker goal would be that one of computing the set of seeds and the corresponding messages so that the probability of victory is merely above a given threshold (so the set of feasible solutions would be larger than in the setting described above). Unfortunately, this objective function does not make the problem easier to be solved. Indeed, not only Theorem 1 holds in

this setting regardless of the threshold, but one may show that, as for the goal of maximizing the probability of victory, the inapproximability still holds when only two candidates are available [14].

3.3.3. Threshold Dynamics

The results provided in Section 3, that are based on a multi-issue independent cascade model, can be extended to settings in which information diffuses according to the *linear threshold model* [15]. This represents the most prominent among the diffusion models alternative to the independent cascade. In the linear threshold model, for each node v of the network, there is a threshold θ_v drawn randomly in $[0, 1]$, and incoming edges (u, v) have a weight $w_{u,v}$ such that $\sum_{(u,v)} w_{u,v} = 1$. Then, a node v becomes active at time t only if the sum of weights of edges coming from active nodes passes the threshold.

It is known that this diffusion model leads to different dynamics with respect to the independent cascade model. Still, we show that our proofs can be adapted. In particular, Theorem 1 and Theorem 4 still hold.

Specifically, for the inapproximability result, we use, in place of SET-COVER, a reduction from VERTEX-COVER. This is the problem of deciding whether, given a graph Z of g nodes and an integer h , there is a subset S of at most h nodes of Z such that every edge of Z has at least one endpoint in S . The reduction is similar to the one described in Theorem 1. Namely, the component G_1 consists of r copies of the graph G . Now by setting $n = r(g - h)$, we let components G_2 and G_3 to have the same number of nodes as in the proof of Theorem 1, except that now the nodes in each component are not arranged as a clique, but as a directed ring (so that a message sent by a node in one component will activate all nodes in that component regardless of their threshold). Notice that, by considering the same initial ranks as in the proof of Theorem 1, the expected margin of victory of c_0 increases by 1 if and only if there is in G a vertex cover of size at most h . Finally, the expected increase in the margin of victory when no vertex cover exists can be made as low as desired by increasing r .

On the other side, it directly follows that the greedy algorithm proposed in Theorem 4 works, with the same approximation factor, even with the linear threshold diffusion model. Indeed, it is known that the influence maximization is a monotonic and submodular function even with this dynamics [15], and it can be observed that this is sufficient to make the proof of Theorem 4 to hold.

3.3.4. Seeds with Different Costs

In our model, we assume that each node can be selected as a seed at the same cost. This can be highly unrealistic. Hence, an extension to our model would be to assume that each node u has a different cost $w(u)$ that should be paid for each message initiated by that node.

Intuitively, this extension makes the election control problem harder. Hence, inapproximability results clearly extend to this setting too. Actually, we can prove that the inapproximability holds even if we restrict to undirected graphs. We can do that by adapting, within the framework of the proof of Theorem 1, the reduction described in [28] for the hardness of the influence maximization

problem, from the problem of finding a vertex cover in a *cubic graph*, i.e., a graph in which every vertex has degree exactly equal to three.⁸

Surprisingly, however, we have that, whenever $\delta \leq B$, a poly-time algorithm returning a constant approximation to the election control problem exists even if the nodes have heterogeneous seeding costs. Indeed, this setting admits a poly-time algorithm for influence maximization returning a $\left(1 - \frac{1}{\sqrt{e}}\right)$ -approximation of the optimal seed set [29]. Then, the arguments of the proof of Theorem 4 immediately prove that this algorithm provides a ρ -approximation for the extension of the election control problem to voters with different costs, where $\rho = \frac{B-\delta+1}{2\delta B} \left(1 - \frac{1}{\sqrt{e}}\right)$.

4. Edge Removal Complexity

We study, in this section, the ECER and IMER problems. All the results provided in this section hold even with unitary score distances. Initially, we focus on the IMER problem when one can only remove edges, as its characterization is useful for the characterization of the ECER problem with two candidates and limited budget. We show that the IMER problem is hard. Our proof reduces from the MAXIMUM-SUBSET-INTERSECTION problem that does not admit any constant-factor approximation polynomial-time algorithm unless $\mathbf{P} = \mathbf{NP}$, as showed in [30].

Definition 9 (MAXIMUM-SUBSET-INTERSECTION (MSI)). *Given a finite set $N = \{z_1, \dots, z_n\}$ of elements, a collection $X = \{x_1, \dots, x_g\}$ of sets with $x_i \subset N$, and a positive integer h , the goal is to find exactly h subsets x_{j_1}, \dots, x_{j_h} whose intersection size $|x_{j_1} \cap \dots \cap x_{j_h}|$ is maximum.*

Theorem 6. *For any constant $\rho > 0$, there is no polynomial-time algorithm returning a ρ -approximation to IMER problem when the budget B is finite, unless $\mathbf{P} = \mathbf{NP}$.*

Proof. We reduce from MSI, showing that a constant-factor approximation algorithm for IMER implies the existence of a constant-factor approximation for MSI, thus having a contradiction unless $\mathbf{P} = \mathbf{NP}$. Given an instance (X, N) of MSI, we build an instance of IMER as follows. For each element z_i , we add $n^2 g^2$ nodes $v_{z_i, j}$ with $j \in \{1, \dots, n^2 g^2\}$. For each set $x_i \in X$, we add two nodes $v_{x_i, 1}, v_{x_i, 2}$ and an edge from $v_{x_i, 1}$ to $v_{x_i, 2}$. All $v_{x_i, 1}$ are seeds, while each $v_{x_i, 2}$ has an edge to each node $v_{z_i, j}, z_i \in N \setminus x_i$ with $j \in \{1, \dots, n^2 g^2\}$, i.e., all the nodes of all the elements not in the set x_i . Figure 4 depicts an example of network built with the above mapping. The budget is set equal to $g - h$.

Notice that, in the optimal solution, only edges from nodes $v_{x_i, 1}$ to $v_{x_i, 2}$ are removed. Thus, the problem reduces to choose $g - h$ sets $x_i \in X$ and

⁸Note that, in that reduction, the author assumes that there is a set of non-allowed seed, that can be simulated in our setting by setting the initial preference rank for those nodes such that c_0 is the best ranked candidate and c_2 the last ranked one.

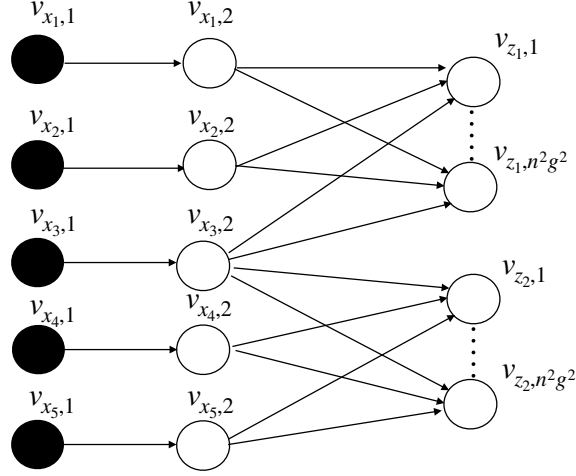


Figure 4: Structure of the election control problem used in the proof of Theorem 6.

remove the edges from $v_{x_i,1}$ to $v_{x_i,2}$, such that as least as possible nodes $v_{z_i,j}$ are influenced. The optimal value is obtained by choosing $X^* \subset X$ of cardinality h that is solution of MSI and then removing the edges from $v_{x_i,1}$ to $v_{x_i,2}$ for all $x \in X \setminus X^*$. Call this set of edges E^* . If we remove edges E^* , all the nodes $v_{z_i,j}, z_i \in \cap_{x_i \in X^*} x_i$ are not influenced, since there are no edges from $v_{x_i,2}, x \in X^*$ to $v_{z_i,j}$, as they all exist in the complement of the bipartite graph.

The relationship between the optimal solution of IMER and that one of MSI is $\Delta I^-(E^*) = g - h + OPT n^2 g^2$, where OPT is the optimal solution to MSI. Assume by contradiction there exists an ρ -approximation algorithm \mathcal{A} for IMER, where $\rho \in (0, 1)$. This implies that there exists an edge set E' such that $\Delta I^-(E') = g - h + APX n^2 g^2$, where APX is an approximation of MSI. Since $\Delta I^-(E') \geq \rho \Delta I^-(E^*)$, then $g - h + APX n^2 g^2 \geq (g - h + OPT n^2 g^2)\rho$, and $APX \geq \frac{(g-h)(\rho-1)}{n^2 g^2} + \rho OPT$. Hence, there exists a ρ' such that $APX \geq \rho' OPT$ and an algorithm \mathcal{A}' for MSI with a ρ' -approximation factor. \square

We can state the following corollary, whose proof directly follows from the proof of the above theorem.

Corollary 1. *For any constant $\rho > 0$, there is not any polynomial time algorithm returning a ρ -approximation to the ECER problem when budget B is finite even when there are two candidates and single-news-article-messages, unless $P = NP$.*

Proof. We can build an instance of ECER with the same graph of Theorem 6, two candidates, all nodes with scores $\langle 1, 0 \rangle$ and seeds with messages $\langle -1, 0 \rangle$. It is easy to see that $\Delta_{\text{MoV}} = 2\Delta I^-$. Since approximating ΔI^- is hard, it follows that approximating Δ_{MoV}^- is hard too. \square

Since with finite budget even the setting with single-news-article messages and two candidates is hard, we focus on those problems in which the budget is unlimited ($B = \infty$). Notice that, while a finite budget corresponds to the case in which a manipulator pays a platform, in the case in which the manipulator is the platform itself, the budget is actually unlimited.

In networks with single-news-article-messages and only two candidates, the optimal solution can be found easily. Intuitively, the problem becomes easy because we can easily solve IMER. If we have unlimited budget, the optimal solution to IMER removes all the edges. It is then easy to extend this solution to solve the ECEA when all the seeds send the same message on the same single candidate and there are only two candidates: if the message is negative for c_0 , *e.g.*, $q_0 = -1$ or $q_1 = 1$, we remove all edges from the network, clearly minimizing the negative effects of the diffusion of the message; if the message is positive for c_0 , *e.g.*, $q_0 = 1$ or $q_1 = -1$, since we cannot increase the diffusion by removing edges, we do not modify the network. From the previous arguments, we can directly state the following.

Observation 1. *There exists a polynomial-time algorithm for the ECER problem with single-news-article messages, two candidates, and unlimited budget.*

Now we show that extending the setting to three or more candidates elections or to the diffusion of different messages makes the problem hard. We introduce the INDEPENDENT-SET problem, that is known not to be approximable to any constant factor in [31], to prove the hardness of the ECER when there are three candidates and messages are single-news-article.

Definition 10 (INDEPENDENT-SET). *Given a graph $G = (X, N)$, with $|X| = g$ vertexes and $|N| = n$ edges, find the largest set of vertexes X^* such that there is no edge connecting two vertexes in X^* .*

Theorem 7. *For any constant $\rho > 0$, there is no polynomial time algorithm returning a ρ -approximation to the ECER with single-news-article messages even when there are three candidates and the budget is unlimited, unless $P = NP$.*

Proof. Given an instance of INDEPENDENT-SET, we build an instance of election control as follows. We add a line L_1 of $ng - g$ nodes with preference $\langle 2, 0, 1 \rangle$ and we seed the first node of the line with a message with $q_0 = q_1 = 0$ and $q_2 = 1$. We add a node v_{x_i} for each node $x_i \in X$ with preferences $\langle 2, 0, 1 \rangle$ and an edge from the last element of the line L_1 to v_{x_i} . For each element $z_i \in N$, we add a line L_{z_i} of g nodes with preferences $\langle 0, 2, 1 \rangle$ and an edge from each $x_j \ni z_i$ to the first node of L_{z_i} . Moreover, we add $n^2 g^2$ isolated nodes with preferences $\langle 2, 1, 0 \rangle$ and $n^2 g^2$ isolated nodes with preferences $\langle 1, 2, 0 \rangle$. Figure 5 depicts an example of network produced with the above mapping. Note that, if no edge is removed, all non-isolated voters change their preferences and vote c_2 , implying $\text{MoV}(S, M, E, H) = 0$. We prove that a constant-factor approximation for ECER would lead to a constant-factor approximation for INDEPENDENT-SET.

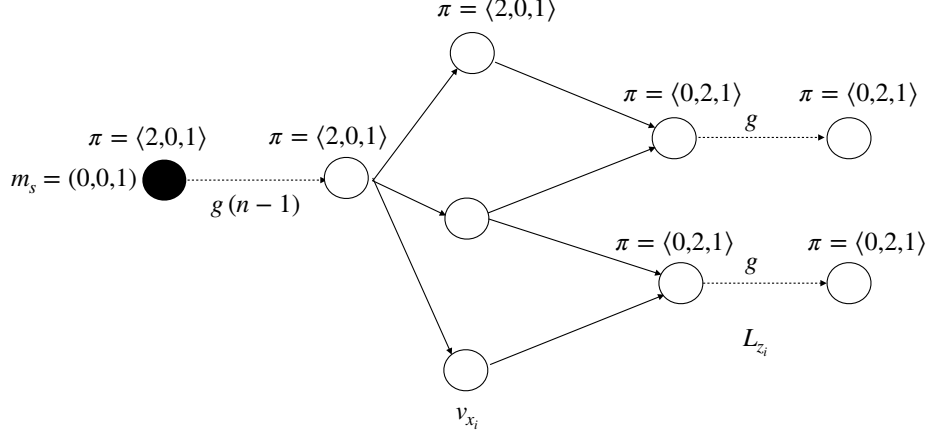


Figure 5: Structure of the election control problem used in the proof of Theorem 7.

Suppose that there exists a set of edges E' , such that $\Delta_{\text{MoV}}^-(E') > 0$. Then c_1 loses all her votes in non-isolated nodes, otherwise $V_{c_0}(S, M, E', H)^* \leq n^2 g^2 - (n-1)g$ and $V_{c_1}^* \geq n^2 g^2 - (n-1)g$. This suggests that the optimal solution is given by the greatest independent set $X^* \subseteq X$. In particular, E^* is given by all the edges from the last node of L_1 to all v_{x_i} with $x_i \in X^*$. Notice that the set of active nodes $v_{x_i}, x_i \in X \setminus X^*$ is the complement of a maximum independent set and hence a minimum vertex cover. Thus, removing all edges in E^* , we obtain $\Delta_{\text{MoV}}^-(E^*) = |X^*|$.

Suppose there exists a ρ -approximation algorithm \mathcal{A} for the ECER problem that removes edges E' . This implies that $\Delta_{\text{MoV}}^-(E') \geq \rho \Delta_{\text{MoV}}^-(E^*)$, where E' is the set of the edges removed by algorithm \mathcal{A} . Since $\Delta_{\text{MoV}}^-(E') > 0$, \mathcal{A} removes only edges from L_1 to v_{x_i} since, if it removes edges between nodes in L_1 , we would have $\Delta_{\text{MoV}}^-(E') \leq 0$. Moreover, all lines L_{z_i} must be active. Hence, the active vertices v_{x_i} are a vertex cover and the inactive vertices in v_{x_i} are an independent set. We remark that the value of $\Delta_{\text{MoV}}^-(E')$ is exactly the number of inactive vertices, *i.e.*, the vertices in the independent set. Thus, if there exists a ρ -approximation algorithm for ECER, there exists a ρ approximation algorithm for INDEPENDENT-SET, leading to a contradiction. \square

We now focus on the case in which messages can be arbitrary and only two candidates. We reduce from the SET-COVER problem to prove the hardness of the ECER problem even in these settings.

Theorem 8. *For any $\rho > 0$, there is no polynomial time algorithm returning a ρ -approximation to the ECER even with two candidates and unlimited budget, unless $\text{P} = \text{NP}$.*

Proof. Consider an instance of SET-COVER. We suppose, w.l.o.g., $n > g$ and build a graph as follows. We add a node v_1 with preferences $\langle 1, 0 \rangle$ and seeded with messages $q_0 = 1$ and $q_1 = -1$, a node v_2 with preferences $\langle 1, 0 \rangle$ and seeded

with message $q_0 = -1$, and an edge between v_1 and v_2 . We add a line L_1 of $n^2 - h - 1$ nodes with preferences $\langle 1, 0 \rangle$ and an edge of probability $\frac{1}{2}$ from v_1 to the first node of the line and an edge from v_2 to the first node of the line. Moreover we seed the first node of the line with message $q_1 = 1$. We add a node v_{x_i} for each set $x_i \in X$ with preferences $\langle 1, 0 \rangle$ and an edge from the last element of the line L_1 to v_{x_i} . For each element $z_i \in N$, we add a line L_{z_i} of n nodes with preference $\langle 0, 1 \rangle$ and an edge from each $x_j \in z_i$ to the first node of L_{z_i} . Moreover, we add $g - h + 1$ isolated nodes with preferences $\langle 0, 1 \rangle$. Figure 6 depicts an example of network produced with the above mapping. Note that, if no edge is removed, all the voters do not change their preferences and $\text{MoV} = 0$. We prove that Δ_{MoV}^- is larger than 0 if and only if SET-COVER is satisfiable.

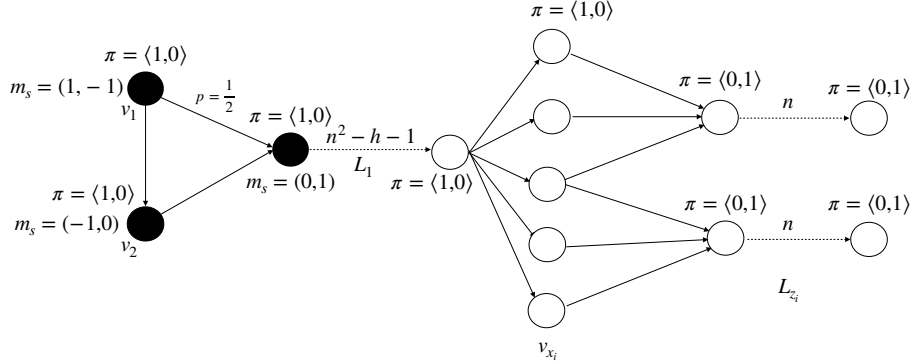


Figure 6: Structure of the election control problem used in the proof of Theorem 8.

If. Define the set of removed edges E^* as composed by the edge between v_2 and L_1 and the incoming edge of each v_{x_i} with $x_i \in X \setminus X^*$. We have two possible live graphs: H_1 if the edge between v_1 and L_1 is active, H_2 otherwise. Thus, $\Delta_{\text{MoV}}^-(E^*, H_1) = 2n^2$ and $\Delta_{\text{MoV}}^-(E^*, H_2) = 2(-n^2 + h + 1 - h) = -2n^2 + 2$. Hence, $\Delta_{\text{MoV}}^-(E^*) = 1$.

Only if. Suppose we do not remove neither the edge from v_1 towards v_2 nor the edge from v_2 towards L_1 . In this case, no voter changes her vote from c_1 to c_0 since all the nodes that votes for c_1 receive messages $q_0 = 1$, $q_0 = -1$, $q_1 = 1$, and $q_1 = -1$. Thus, one of the two aforementioned edges should be removed. It easy to see that removing the edge from v_2 to L_1 is the best choice. Since c_0 must take some of the votes of c_1 , the message in v_1 must reach at least some lines in L_z and no edges must be removed in L_1 . We have two possible live graphs: H_1 if the edge between v_1 and L_1 is active, H_2 otherwise. Assume by contradiction that in H_1 not all lines L_z vote for c_0 . This implies that $\Delta_{\text{MoV}}^-(E^*) \leq \frac{2(n(n-1)-2(n^2+h+1))}{2} < 0$, where E^* is the set of removed edges. Hence, in H_1 , all line L_z must be active and $\Delta_{\text{MoV}}^-(E^*, H_1) = 2n^2$. In H_2 , $\Delta_{\text{MoV}}^-(E^*)$ must be larger than $-2n^2 + 1$ and at most h nodes v_{z_i} can be active. Thus, there exists a set cover of size h . \square

5. Edge Addition Complexity

We study, in this section, the ECEA and IMEA problems. All the results provided in this section hold even with unitary score distances. Initially, we study the complexity of IMEA problem with a finite budget. First, we notice that the APX-hardness of the IMEA problem directly follows from the APX-hardness of the influence maximization problem by seeding. In fact, the seeding problem with network $G(V, E, p)$ and budget B is equivalent to the edge-addition problem with the same graph, except for an additional isolated node v_1 , that is the only seed, in which we can add at most B edges and the probabilities p of the new (added) edges are all zero except for the edges connecting v_1 to the nodes of V , whose probabilities p are one. We improve this result, showing that the IMEA problem is harder to approximate than influence maximization by seeding. Indeed, IMEA cannot be approximated to any constant factor, unless $P = NP$, while influence maximization by seeding can be. In our proof, we reduce from the maximization version of SET-COVER, called MAX-COVER.

Definition 11 (MAX-COVER). *Given a finite set $N = \{z_1, \dots, z_n\}$ of elements, a collection $X = \{x_1, \dots, x_g\}$ of sets with $x_i \subset N$, and $h \in \mathbb{N}^+$, the objective is to select $X^* \subset X$, with $|X^*| \leq h$, that maximizes $|\cup_{x_i \in X^*} x_i|$.*

Feige proves that deciding whether in an instance of MAX-COVER all the elements can be covered or at most a $(1 - \frac{1}{e} + \epsilon)$ fraction of them is NP-hard for any $\epsilon > 0$ [32].

Theorem 9. *For any constant $\rho > 0$, there is no polynomial time algorithm returning a ρ -approximation to the IMEA problem when B is finite, unless $P = NP$.*

Proof. Consider an instance of MAX-COVER. We assume, w.l.o.g., $g < n$ and we build an instance of IMEA as follows: for each i in $\{1, \dots, n^8\}$, we add a node v_i , a node v_{i,x_j} for each $x_j \in X$ and a node v_{i,z_t} for each $z_t \in N$. Moreover, we add an edge from each v_{i,x_j} to each $v_{i,z_t}, z_t \in x_j$ with probability 1 and an edge from v_{i,z_t} to v_{i+1} with probability $1 - \frac{1}{n^{\frac{8}{n}}}$. We add a node v_{n^8+1} and an edge with probability 1 towards n^{10} nodes. Call the subgraph composed by these nodes G' . The resulting graph is depicted in Figure 7. The only seed is v_1 , and the only edges that can be added are the edges between v_i and $v_{i,x_j}, x_j \in X$ with probability one. The budget is hn^8 . If MAX-COVER is satisfiable, *i.e.*, there exists a set X^* that covers all the elements, there exists a solution E^* to IMEA in which for each $i \in \{1, \dots, n^8\}$ we add the edges from v_i to h v_{i,x_j} such that if v_i is active then all v_{i,z_t} are active. In this case, $\Delta I^+(E^*)$ is larger than the expected influence on the subgraph G' , *i.e.*, $\Delta I^+ > [1 - (\frac{1}{n^{\frac{8}{n}}})^n] n^8 n^{10} \geq [1 - \frac{1}{n^8}] n^8 n^{10} > (\frac{1}{e} - \epsilon) n^{10}$ for all $\epsilon > 0$ and n large enough. Suppose each cover of size at most h cover at most $\frac{3}{4}$. It implies that at least $\frac{n^8}{h+1}$ nodes v_i have at most h outwards edges and thus they leave at least $\frac{1}{4}$ vertexes v_{i,z_t} without incoming edges. Thus the probability of activating G' is

smaller than $[1 - (\frac{1}{n})^n]^{\frac{hn^8}{h+1}} [1 - (\frac{1}{n})^{\frac{3n}{4}}]^{\frac{n^8}{h+1}} \leq \frac{1}{e^{\frac{h}{h+1}}} \frac{1}{e^{\frac{n^2}{h+1}}} = \frac{1}{e^{\frac{h+n^2}{h+1}}} \leq e^{-n}$. and $\Delta I^+(E') \leq n^8(n+g+1) + 1 + e^{-n}n^{10}$. Clearly $\frac{\Delta I^+(E')}{\Delta I^+(E^*)} < \rho$, for each $\rho > 0$ and n sufficiently large. Hence, a ρ -approximation algorithm for IMEA implies that we can distinguish between satisfiable instances of MAX-COVER and instances in which at most $\frac{3}{4}$ of the elements are covered, leading to a contradiction. \square

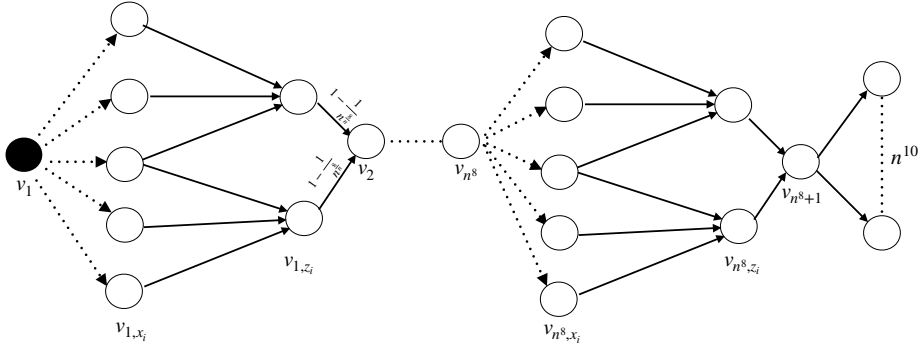


Figure 7: Structure of the election control problem used in the proof of Theorem 9.

Now we can state the following result, whose proof is direct from the proof of the above theorem.

Corollary 2. *For any constant $\rho > 0$, there is not any polynomial time algorithm returning a ρ -approximation to the ECEA when B is finite even when there are two candidates and single-news-article messages, unless $P = NP$.*

Thus, we focus on the case with unlimited budget. Since the maximum influence is reached when the network is fully connected, the optimal solution to IMEA with unlimited budget adds all the non-existing edges to the network and thus can be computed in polynomial time. An argument similar to the one used for edge removal shows that ECEA with unlimited budget, two candidates, and single-news-article messages is easy. In particular, if the message is positive for c_0 , *i.e.*, $q_0 = 1$ or $q_1 = -1$, we aim at maximizing the diffusion of the message and we add all the edges. If the message is negative for c_0 , *i.e.*, $q_0 = -1$ or $q_1 = 1$, we aim at minimizing the diffusion and we do not remove any edge. From the previous arguments, we can directly state the following.

Observation 2. *There exists a polynomial-time algorithm for the ECEA problem with single-news-article messages, two candidates and unlimited budget.*

Next, we prove that increasing the number of candidates or allowing arbitrary messages makes the problem hard.

Theorem 10. *For any $\rho > 0$, there is not any polynomial time algorithm returning a ρ -approximation to the ECEA with single-news-article messages even when there are three candidates and the budget is unlimited, unless $P = NP$.*

Proof. Given an instance of SET-COVER, we build an instance of election control as follows. We add a node v_1 with preferences $\langle 0, 1, 2 \rangle$ and seed it with $q_2 = 1$. We add a line L_1 of $ng - h - 1$ nodes with preferences $\langle 2, 0, 1 \rangle$. We add a node v_{x_i} for each set $x_i \in X$ with preferences $\langle 2, 0, 1 \rangle$. For each element $z_i \in N$, we add a line L_{z_i} of g nodes with ranking $\langle 0, 2, 1 \rangle$ and an edge from each $x_j \ni z_i$ to the first node of L_{z_i} . Moreover, we add n^2g^2 isolated nodes with preferences $\langle 2, 1, 0 \rangle$ and n^2g^2 isolated nodes with preferences $\langle 1, 2, 0 \rangle$. An example of network produced with the above mapping is depicted in Figure 8. The only edges that can be added are the node from v_1 to L_1 and the edges from L_1 to each nodes v_{x_i} , *i.e.*, these edges have probability 1 and all other non existing edges have probability 0. Notice that if no edge is added, all nodes will not change their votes, implying $\Delta_{\text{MoV}}(\emptyset) = 0$. We prove that there exists a set $E^* \subseteq E$ with $\Delta_{\text{MoV}}^+(E^*) > 0$ if and only if SET-COVER is satisfiable.

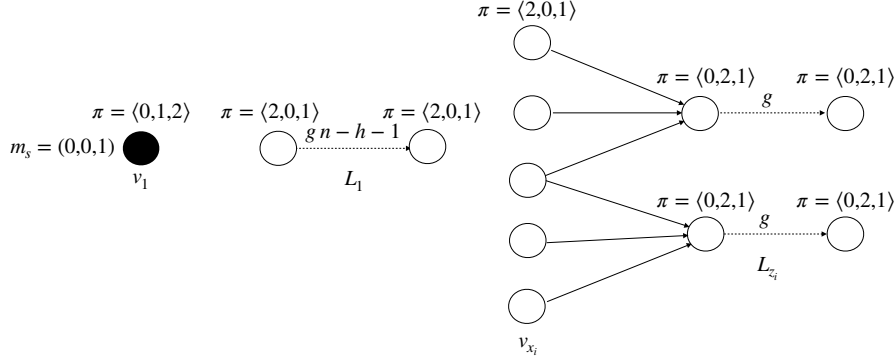


Figure 8: Structure of the election control problem used in the proof of Theorem 10.

If. Given a set cover X^* , define as E^* the set of the edge from v_1 to L_1 and all the edges from L_1 to $v_{x_i}, x_i \in X^*$. If we add edges E^* , c_0 loses $ng - h - 1 + h$ votes, while c_1 loses ng votes, and thus $\Delta_{\text{MoV}}^+(E^*) = 1$.

Only if. The existence of a set E^* with $\Delta_{\text{MoV}}^+(E^*) > 0$ implies that the edge from v_1 to L_1 is added and c_0 loses at least $ng - h - 1$ votes. Thus, c_1 must lose at least $ng - h$ votes, implying that she loses all the not isolated nodes. Since $\Delta_{\text{MoV}}^+(E^*) > 0$, c_0 can lose at most h nodes v_{x_i} , *i.e.*, there are at most h edges from L_1 to v_{x_i} in E^* . Hence, there are h nodes v_{x_i} that cover all the elements z_i and SET-COVER is satisfiable. \square

Theorem 11. For any $\rho > 0$, there is not any polynomial time algorithm returning a ρ -approximation to the ECEA even with two candidates and unlimited budget, unless $\text{P} = \text{NP}$.

Proof. Consider an instance of SET-COVER. We suppose, w.l.o.g., $n > g$ and build a graph as follow. We add a node v_1 with preferences $\langle 1, 0 \rangle$ and seeded with messages $q_0 = 1$ and $q_1 = -1$, and a node v_2 with preferences $\langle 1, 0 \rangle$ and seeded with message $q_1 = 1$. Moreover we add an edge with probability $\frac{1}{2}$ between v_1 and v_2 . We add a line L_1 of $n^2 - h - 1$ nodes with preferences $\langle 1, 0 \rangle$.

We add a node v_{x_i} for each set $x_i \in X$ with preferences $\langle 1, 0 \rangle$. For each element $z_i \in N$, we add a line L_{z_i} of n nodes with preferences $\langle 0, 1 \rangle$ and an edge from each $x_j \ni z_i$ to the first node of L_{z_i} . Moreover, we add $g - h + 1$ isolated nodes with preferences $\langle 0, 1 \rangle$. The edges that can be added are: the edge from v_2 to the first node of L_1 with probability 1 and edges from the last node of L_1 to all v_{x_i} with probability 1 (all other non-existing edges have probability 0).

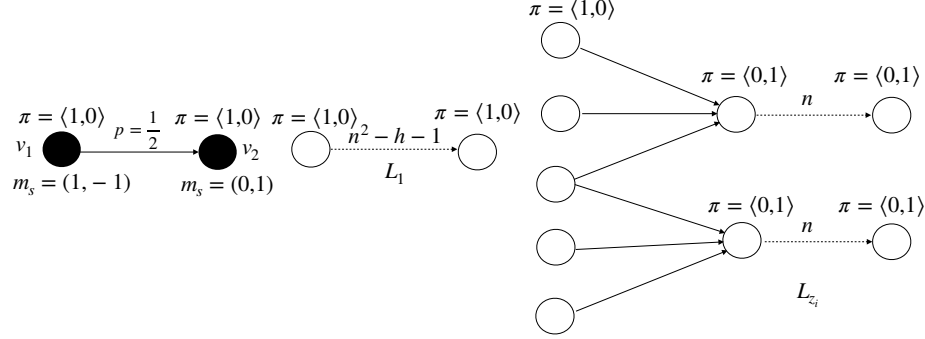


Figure 9: Structure of the election control problem used in the proof of Theorem 11.

Notice that, if no edges are added, no voter changes her votes and $\text{MoV}(\emptyset)$ is 0. We prove that there exists a set $E^* \subseteq E$ with $\Delta_{\text{MoV}}^+(E^*) > 0$ if and only if SET-COVER is satisfiable.

If. The set of added edges E^* is composed by the edge between v_2 and L_1 and the incoming edge of each v_{x_i} with $x_i \in X^*$. We have two possible live graphs: H_1 if the edge between v_1 and v_2 is active, H_2 otherwise. $\Delta_{\text{MoV}}^+(E', H_1) = 2n^2$ and $\Delta_{\text{MoV}}^+(E', H_2) = 2(-n^2 + h + 1 - h) = -2n^2 + 2$. Thus, $\Delta_{\text{MoV}}^+(E') = 1$.

Only if. Suppose we do not add the edge between v_2 and L_1 . In this case, MoV does not change and $\Delta_{\text{MoV}}^+ = 0$. Thus, the edge between v_2 and L_1 must belong to the set of added edges E^* . We have two possible live graphs: H_1 if the edge between v_1 and v_2 is active, H_2 otherwise. Assume by contradiction that in H_1 not all lines L_{z_i} vote for c_0 . This implies that $\Delta_{\text{MoV}}^+(E^*) \leq \frac{2[n(n-1)] - 2[n^2 + h + 1]}{2} < 0$. Hence, in H_1 , all line L_z must be active and $\Delta_{\text{MoV}}^+(E^*, H_1) = 2n^2$. In H_2 , $\Delta_{\text{MoV}}^+(E^*)$ must be larger than $-2n^2 + 1$ and at most h nodes v_{z_i} can be active, *i.e.*, at most h edges from L_1 to voters v_{x_i} can be added. Thus, there exists a set cover of size h , leading to a contradiction. \square

6. Reoptimization Complexity

In this section, we show a form of robustness of our hardness results. Specifically, we consider the following reoptimization setting.

Definition 12. An election control through social influence by seeding reoptimization problem $\text{ReOpt}(I, S^*, e, o)$ is defined as follows.

- *INPUT*: (I, S^*, e, o) , where I is an instance of election control, S^* is an optimal solution to I , $e \in V \times V$ is an edge and $o \in [0, 1]$ is a probability.
- *OUTPUT*: the optimal solution to I_1 , where I_1 is obtained changing the probability of edge e to o in the instance I .

We prove that Theorem 1 can be extended to prove the hardness of reoptimization.

Theorem 12. *For any $\rho > 0$ even depending on the size of the problem, there is not any poly-time algorithm returning a ρ -approximation to the reoptimization problem for ECS, unless $P = NP$.*

Proof. Consider the reduction in Theorem 1. We build an instance I of ECS in which we replace the set cover instance in G_1 with the following graph. For each $z_i \in Z$, there is a node v_{z_i} . For each $x_i \in X$, we add two nodes $v_{x_{i,1}}$ and $v_{x_{i,2}}$, and an edge from $v_{x_{i,1}}$ to $v_{x_{i,2}}$. Moreover, we add an edge from $v_{x_{i,2}}$ to all v_z , $z \in x_i$. Finally, we add a node v^* with an edge from v^* to all nodes v_z , $z \in Z$, and an edge from v^* to $v_{x_{1,2}}$ (or any node $x_{i,2}$). We modify the graphs G_2 and G_3 in such a way that c_1 needs the votes of $2h + n$ nodes of G_1 . Let S^* be the optimal solution of I that includes seeds v_z , any h nodes $v_{x_{i,1}}$ and a node in G_2 . Consider the problem $ReOpt(I, S^*, (v^*, v_{x_1}), 0)$, its optimal solution is the optimal solution of the optimization problem over I . If $ReOpt(I, S^*, (v^*, v_{x_1}), 0)$ can be approximated in polynomial time, then set cover can be solved in polynomial time. \square

Similarly, we consider the reoptimization problems for edge removal or edge addition.

Definition 13. *An election control through social influence by edge removal or addition reoptimization problem $ReOpt(I, E^*, e, o)$ is defined as follows.*

- *INPUT*: (I, E^*, e, o) , where I is an instance of election control, E^* is an optimal solution to I , $e \in V \times V$ is an edge and $o \in [0, 1]$ is a probability.
- *OUTPUT*: the optimal solution to I_1 , where I_1 is obtained changing the probability of edge e to o in the instance I .

The following theorem shows a general result, that extends the hardness of the optimization problem to its reoptimization variant whenever a simple condition is satisfied.

Theorem 13. *For the set of election control problems by edge removal or addition with $\max_v \{\max_{c_i} \pi_v(i) - \min_{c_i} \pi_v(i)\} = O(\text{poly}(\text{size}(I)))$, reoptimization is as hard as optimization.*

Proof. Consider an instance I of election control with $G = (V, E, p)$. By assumption $d = \max_v [\max_{c_i} \pi_v(i) - \min_{c_i} \pi_v(i)] = O(\text{poly}(\text{size}(I)))$, i.e., d is polynomially upper bounded in the instance size. We build a graph G_1 with

$d + 1$ nodes $\{v_i\}, i \in \{0, \dots, d\}$ with seeds $q_0 = 1$. We add a node v_1^* with an edge from each node in v_i to v_1^* . We add a node v_2^* with an edge from v_1^* to v_2^* . Moreover, we add an edge from v_2^* to any node of G . In edge addition instances, we set $p = 0$ for all (non-existing) edges among v_i and G . Finally, we set the preferences of v_0 and v_i s.t. they will vote for c_0 , i.e., $\pi(0) > \pi(i)$ holds for every $c_i \neq c_0$.

Notice that, since all nodes in G receive $d + 1$ positive messages on c_0 and c_0 is loosing by at most d in each preferences, all nodes will vote for c_0 . Thus the optimal solution removes/adds no edges. Consider the problem $ReOpt(I, \emptyset, (v_1^*, v_2^*), 0)$, its optimal solution is the optimal solution of the optimization problem over I . \square

In the reductions used in the proofs of all the theorems provided in the previous sections, $\max_v \{\max_{c_i} \pi_v(i) - \min_{c_i} \pi_v(i)\}$ is constant. Hence, as a corollary of Theorem 13, we have that all our hardness results on optimization problems extend to their reoptimization variants.

7. Conclusions and Future Work

In this work, we analyze the problem of manipulating the result of an election (a.k.a. election control through social influence) by some forms of manipulations. More precisely, we investigate both the case in which the manipulator can make seeding and the case in which the manipulator can alter the network by removing or adding edges. We prove a tight characterization of the settings in which computing an approximation to the best manipulation can be infeasible or feasible. In particular, our results show that, except for trivial classes of instances, the manipulation is hard, even when one accepts an approximation of the margin of victory. In particular, we provide a comprehensive study, investigating various subcases to identify the minimal subsets of instances for which the election control problem is hard. We show that manipulation by seeding is hard even with basic instances such as line graphs and that the most known greedy algorithms for social influence do not provide any approximation factor even with basic graphs. In the case of edge removal or addition, we also show that, even when the manipulator has an unlimited budget, the problem is hard. Interestingly, we derive a similar result also to influence maximization/minimization, as this problem was unexplored so far. Finally, we show that our hardness results hold in a reoptimization variant. This bundle of results is, therefore, positive for the democracy, as it is unlikely that a manipulator can effectively manipulate an election.

While we provided a poly-time constant-approximation algorithm in many settings, we did not try to optimize the approximation ratio. Hence, it would be interesting to design algorithms that can improve on ours, and close the small gaps existing among our results. Furthermore, it would be interesting to analyze other generalizations of our model, e.g., different models for information diffusion and time-evolving networks.

References

- [1] M. Castiglioni, D. Ferraioli, N. Gatti, Election control in social networks via edge addition or removal, in: AAAI, 2020.
- [2] H. Allcott, M. Gentzkow, Social media and fake news in the 2016 election, *Journal of economic perspectives* 31 (2) (2017) 211–36.
- [3] A. Guess, B. Nyhan, J. Reifler, Selective exposure to misinformation: Evidence from the consumption of fake news during the 2016 us presidential campaign, *European Research Council* 9.
- [4] E. Ferrara, Disinformation and social bot operations in the run up to the 2017 french presidential election, *First Monday* 22 (8).
- [5] A. Alaphilippe, C. Ceccarelli, L. Charlet, M. Mycielski, Disinformation detection system: 2018 italian elections, Brussels: EU Disinfo Lab, June 1.
- [6] F. Giglietto, L. Iannelli, L. Rossi, A. Valeriani, N. Righetti, F. Carabini, G. Marino, S. Usai, E. Zurovac, Mapping italian news media political coverage in the lead-up to 2018 general election, Available at SSRN 3179930.
- [7] S. Sina, N. Hazon, A. Hassidim, S. Kraus, Adapting the social network to affect elections, in: AAMAS, 2015, pp. 705–713.
- [8] V. Auletta, I. Caragiannis, D. Ferraioli, C. Galdi, G. Persiano, Minority becomes majority in social networks, in: WINE, 2015, pp. 74–88.
- [9] V. Auletta, I. Caragiannis, D. Ferraioli, C. Galdi, G. Persiano, Information retention in heterogeneous majority dynamics, in: WINE, 2017, pp. 30–43.
- [10] V. Auletta, I. Caragiannis, D. Ferraioli, C. Galdi, G. Persiano, Robustness in discrete preference games, in: AAMAS, 2017, pp. 1314–1322.
- [11] V. Auletta, D. Ferraioli, G. Greco, Reasoning about consensus when opinions diffuse through majority dynamics., in: IJCAI, 2018, pp. 49–55.
- [12] V. Auletta, D. Ferraioli, V. Fionda, G. Greco, Maximizing the spread of an opinion when tertium datur est, in: AAMAS, 2019, pp. 1207–1215.
- [13] R. Brederick, E. Elkind, Manipulating opinion diffusion in social networks, in: IJCAI, 2017.
- [14] B. Wilder, Y. Vorobeychik, Controlling elections through social influence, in: AAMAS, 2018, pp. 265–273.
- [15] D. Kempe, J. Kleinberg, E. Tardos, Maximizing the spread of influence through a social network, in: ACM SIGKDD, 2003, pp. 137–146.

- [16] F. Corò, E. Cruciani, G. D’Angelo, S. Ponziani, Vote for me!: Election control via social influence in arbitrary scoring rule voting systems, in: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS ’19, Montreal, QC, Canada, May 13-17, 2019, 2019, pp. 1895–1897.
- [17] F. Corò, E. Cruciani, G. D’Angelo, S. Ponziani, Exploiting social influence to control elections based on scoring rules, in: IJCAI, 2019, pp. 201–207.
- [18] M. Aboueimehrizi, F. Corò, E. Cruciani, G. D’Angelo, Election control with voters’ uncertainty: Hardness and approximation results, CoRR abs/1905.04694.
- [19] R. Becker, F. Corò, G. D’Angelo, H. Gilbert, Balancing spreads of influence in a social network, CoRR abs/1906.00074.
- [20] V. Auletta, D. Ferraioli, V. Savarese, Manipulating an election in social networks through edge addition, in: AIIA, 2019.
- [21] D. Sheldon, B. Dilkina, A. N. Elmachtoub, R. Finseth, A. Sabharwal, J. Conrad, C. Gomes, D. Shmoys, W. Allen, O. Amundsen, W. Vaughan, Maximizing the spread of cascades using network design, in: Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, UAI’10, 2010, pp. 517–526.
- [22] E. B. Khalil, B. Dilkina, L. Song, Scalable diffusion-aware optimization of network topology, in: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’14, 2014, pp. 1226–1235.
- [23] M. Kimura, K. Saito, H. Motoda, Solving the contamination minimization problem on networks for the linear threshold model, 2008, pp. 977–984.
- [24] C. J. Kuhlman, G. Tuli, S. Swarup, M. V. Marathe, S. S. Ravi, Blocking simple and complex contagion by edge removal, in: 2013 IEEE 13th International Conference on Data Mining, 2013, pp. 399–408.
- [25] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti Spaccamela, M. Protasi, Complexity and approximation: Combinatorial optimization problems and their approximability properties, Springer Science & Business Media, 2012.
- [26] A. Wang, W. Wu, L. Cui, On bharathi–kempe–salek conjecture for influence maximization on arborescence, Journal of Combinatorial Optimization 31 (4) (2016) 1678–1684.
- [27] P. Manurangsi, Almost-polynomial ratio eth-hardness of approximating densest k-subgraph, in: Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017, 2017, pp. 954–961.

- [28] S. Khanna, B. Lucier, Influence maximization in undirected networks, in: ACM-SIAM SODA, 2014, pp. 1482–1496.
- [29] H. Nguyen, R. Zheng, On budgeted influence maximization in social networks, IEEE Journal on Selected Areas in Communications 31 (6) (2013) 1084–1094.
- [30] On the inapproximability of maximum intersection problems, Information Processing Letters 112 (19) (2012) 723 – 727.
- [31] D. Zuckerman, Linear degree extractors and the inapproximability of max clique and chromatic number, Theory of Computing 3 (6) (2007) 103–128.
- [32] U. Feige, A threshold of $\ln n$ for approximating set cover, J. ACM 45 (4) (1998) 634–652.

Appendix A. Omitted Proofs

Proposition 1. *For any $\rho > 0$ even depending on the size of the problem, no algorithm following the greedy approach returns a ρ -approximation to the ECS problem, even in undirected graphs in which each node has degree at most 2.*

Proof. Consider the graph given in Figure A.10: it is composed of three graphs G_1, G_2, G_3 , and both G_2 and G_3 are composed of two subgraphs, respectively, $G_{2,1}, G_{2,2}$ and $G_{3,1}, G_{3,2}$. Furthermore, the graph is undirected and each node has a degree of at most 2. The influence probability associated with every edge is one. According to the preference ranks of the nodes, candidate c_2 collects 5 votes, while candidates c_1 and c_0 gather 7 votes each. Thus, the actual margin of victory of c_0 is equal to zero. Suppose $B = 2$.

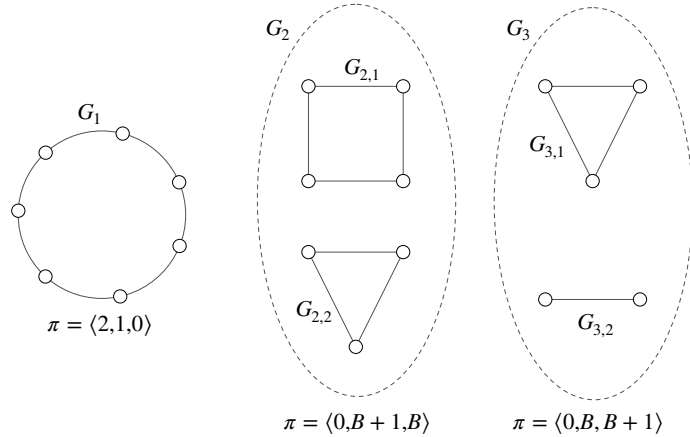


Figure A.10: example of a small undirected network in which the greedy algorithm performs badly.

Since, for all the nodes of G_2 and G_3 , the score difference between c_0 and the most-preferred candidate is larger than the budget B and $B = 2$, it is clear that c_0 cannot get any further vote. Then, to increase the margin of victory of c_0 , it is necessary that c_2 obtains some of the c_1 's votes. The optimal solution (S^*, M^*) is that, while c_0 keeps 7 votes, c_1 and c_2 collect 6 votes each, providing $\mathbb{E}_H[\text{MoV}(S^*, M^*, E, H)] = 1$. This can be obtained by setting $m_v(2) = 1$ for a single $v \in G_{2,2}$ and $m_v(1) = 1$ for a single $v \in G_{3,2}$.

However, this solution cannot be found by any algorithm adopting the greedy approach described above. Indeed, we next show that $\mathcal{F}(\emptyset, ())$ is empty, and thus the algorithm never adds any seed in S : clearly, $\mathcal{F}(\emptyset, ())$ cannot contain any pair (s, m_s) that increases the number of votes of c_0 ; moreover, by seeding a node in G_1 the margin of victory clearly cannot increase (it either remains unchanged, or it decreases if c_0 ceases to be the best ranked candidate); similarly, by seeding a node in G_2 , either the margin of victory reduces (if c_2 passes c_1) or remains unchanged; finally, by seeding one node in G_3 either the margin of victory reduces (if c_1 passes c_2) or remains unchanged.

Hence, the greedy solution results in a zero margin of victory, and thus it cannot be a ρ -approximation. \square

Proposition 2. *For any $\rho > \frac{38}{|V|}$, no algorithm following the greedy approach returns a ρ -approximation to the ECS problem, even in directed trees.*

Proof. Consider the graph in Figure A.11. It is composed of 5 subgraphs A_1, A_2, A_3, A_4, A_5 , where A_1 is a directed line of 7 r nodes, while A_2, A_3, A_4, A_5 are directed trees in which there is a root and the remaining nodes are children of the root. The specific number of nodes of every subgraph is reported in the figure. The influence probability associated with every directed edge is one. Let $r > \frac{2}{\rho}$. Observe that $|V| = 19r$ and therefore $\rho > \frac{38}{|V|}$. According to the preference ranks of the nodes, the candidate c_2 collects $5r$ votes, while candidates c_1 and c_0 gather $7r$ votes each. Then, the actual margin of victory c_0 is equal to zero. Suppose $B = 2$.

Due to the budget constraint, c_0 cannot get any further vote, and thus, to increase the margin of victory of c_0 , we need that c_2 gets some of the c_1 's votes. The optimal solution (S^*, M^*) is then obtained by setting that $S^* = \{x, y\}$ (see in the figure which nodes are labeled as x and y) has m_x with $m_x(2) = 1$ and m_y with $m_y(1) = 1$, and the expected margin of victory is r . However, this solution cannot be found by any algorithm adopting the greedy approach. Indeed, we have that no pair (s, m_s) can increase the number of votes of c_0 ; moreover, by seeding a node in A_1 the margin of victory clearly cannot increase (it either remains unchanged, or it decreases if c_0 ceases to be the best ranked candidate); by seeding a node in A_4 or in A_5 , either the margin of victory decreases (if c_2 passes c_1) or remains unchanged; finally, by seeding the root of A_1 or the root of A_2 , either the margin of victory decreases (if c_2 passes c_1) or remains unchanged. Hence, the only action that the greedy algorithm can take would be to select as seed either a leaf of A_1 , or a leaf of A_2 and letting them to change its vote from c_1 to c_2 . By repeating the argument, we have that the

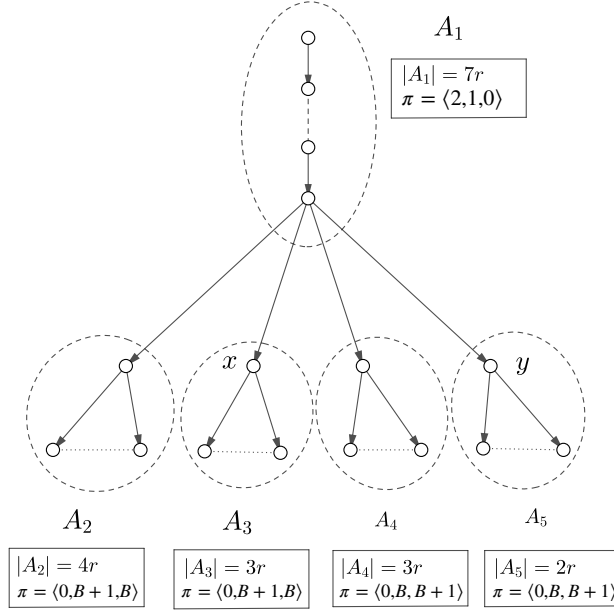


Figure A.11: Example of a tree in which the greedy algorithm performs badly.

two seeds selected by a greedy algorithm must be two leaves from $A_1 \cup A_2$. So, the expected margin of victory is 2, and the approximation factor is $\frac{2}{r} < \rho$. \square

Theorem 2. *The ECS problem with at least four candidates is NP-hard even on line graphs.*

Proof. We reduce from PARTITION. This problem, given a set of positive integers $A = \{a_1, a_2, \dots, a_n\}$, asks if there is a subset $K \subset A$ whose sum is equal to t , where $t = \sum_{a \in A} a/2$. It is easy to see that the NP-hardness of this problem holds even imposing that the cardinality of S is $k \leq n/2$.

Hence, given a set A of positive integers, a target t and a number k as the input of PARTITION, we build a graph as in Figure A.12. There are four candidates c_0, c_1, c_2, c_3 and budget $B = k$. Some voters are isolated nodes, while the others are arranged in n independent lines.⁹ There are $5n$ isolated nodes with preference rank $\langle 0, B+2, B+1, B \rangle$, $8n+1-8k$ isolated nodes with preference rank $\langle 0, B+1, B+2, B \rangle$ and $3n$ nodes with preference rank $\langle 4, 3, 2, 1 \rangle$. Each line is composed by 5 nodes. For each line $i \in \{1, 2, \dots, n\}$, the first three nodes have preference rank $\langle 0, B+2, B+1, 1 \rangle$, the last two nodes have preference rank $\langle 0, B, B+1, B+2 \rangle$, the edges connecting $v_{a,i}$ to $v_{b,i}$ and $v_{d,i}$ to $v_{e,i}$ have probability 1, and the edges $(v_{b,i}, v_{c,i})$ and $(v_{c,i}, v_{d,i})$ are activated,

⁹The graph can be seen as a single line with zero probability links going from $v_{e,j}$ to $v_{a,j+1}$, for $j = 1, 2, \dots, n-1$, and connecting the isolated nodes.

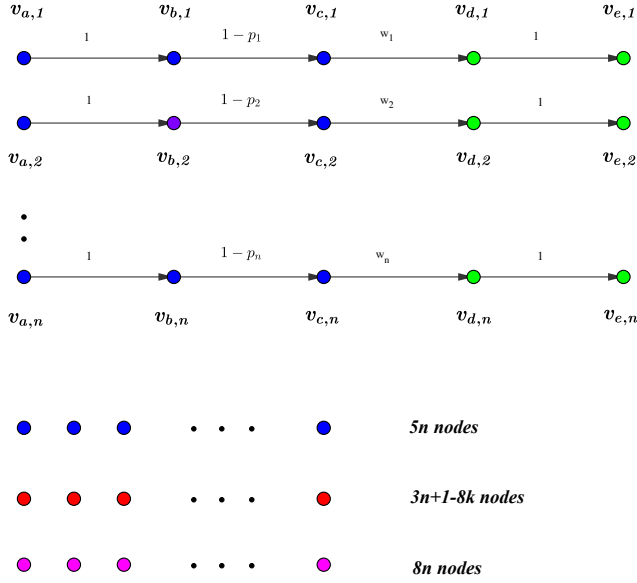


Figure A.12: Structure of the election control problem used in the proof of Theorem 2. Blue nodes represent voters with preference rank $\langle 0, B+2, B+1, 1 \rangle$, green nodes represent voters with preference rank $\langle 0, B, B+1, B+2 \rangle$, while red nodes and pink nodes have preference rank, respectively, $\langle 0, B+1, B+2, B \rangle$, and $\langle 4, 3, 2, 1 \rangle$.

respectively, with probability $1-p_i$ and w_i , where:

$$p_i = \frac{a_i}{4t}, \quad w_i = \frac{2^{-4p_i}}{(1-p_i)(2 \ln 2)^{1/k}}.$$

Note that, since $a_i < t, \forall i \in \{1, \dots, n\}$, then $p_i < \frac{1}{4}, \forall i$. It is easy to see that, since $p_i < \frac{1}{4}$, it holds $w_i < 1, \forall i$. c_1 collects $8n$ votes, c_2 collects $8n+1-8k$ votes, c_3 collects $2n$ votes, and c_0 collects $8n$ votes. Hence, $\text{MoV}(\emptyset, (), E, H) = 0$.

Since c_0 cannot gain votes, the only way to increase the margin of victory is making c_1 lose some of her votes in favor of the other candidates. First, we prove that the optimal solution is given by a set of k nodes $v_{a,i}$ with messages $m_s(2) = 1$ (or equivalently $m_s(1) = -1$). Consider any seed set S' of k nodes $v_{a,i}$ and let $K' = \{i : v_{a,i} \in S'\}$. Notice that c_1 has more votes than c_2 unless all the edges among $v_{b,i}, v_{c,i}$ and $v_{d,i}$ are active for all $i \in K'$. This happens with probability $\prod_{i \in K'} (1-p_i)w_i$. In this case c_1 has $8n-3k$ votes and c_2 has $8n+1-3k$. When at least an edge is not active, the MoV is given by the number of nodes lost by c_1 . Hence, the expected MoV is

$$\begin{aligned} \mathbb{E}_H[\Delta_{\text{MoV}}^S(S', M, H)] &= 2k + \prod_{i \in K'} (1-p_i) - \prod_{i \in K'} (1-p_i)w_i \\ &= 3k - \prod_{i \in K'} p_i - \prod_{i \in K'} (1-p_i)w_i. \end{aligned}$$

Suppose the optimal solution S^* takes some seeds $S_1^* \subseteq S^*$ not in the set $\{v_{a,i}\}$. Then, for each seed $s \in S_1^*$, the expected number of votes of c_1 decreases of at most $1 + (1 - p_i) = 2 - p_i$. Define the set $R(z)$ as the set of the z smallest p_i . Consider the set $R(k)$ and the set of seeds $S'' = \{v_{a,i}\}_{i \in R(k)}$. $\mathbb{E}_H[\Delta_{\text{MoV}}^S(S'', M, H)] = 3k - \prod_{i \in R(k)} p_i - \prod_{i \in R(k)} (1 - p_i) w_i > 3k - 1 - \prod_{i \in R(k)} p_i$, where the last inequality follows from $\prod_{i \in R(k)} (1 - p_i) w_i < 1$. It follows that $\mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M, H)] \leq |S_1^*|(2 - p_i) + \sum_{i \in R(|S_1^*| - |S_1^*|)} (3 - p_i) < \mathbb{E}_H[\Delta_{\text{MoV}}^S(S'', M, H)]$.

We proved that in the optimal solution S^* , all the seeds are placed at the beginning of the line. Let $K^* = \{i : v_{a,i} \in S^*\}$ and let $x = \sum_{i \in K^*} p_i$. Then the derivative of $\mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)]$ with respect to x is equal to $-1 + 2^{-4x+1}$. This means that the value of x that maximizes the margin of victory is $\frac{1}{4}$, which is equivalent to $\sum_{i \in K^*} a_i = t$. This holds if and only if we reduce from a "yes" instance of partition. Hence a polynomial time algorithm for the ECS problem, would allow us to solve PARTITION in polynomial time, leading to a contradiction unless $P=NP$. \square

Theorem 3. *If there is a $\rho > 0$ approximation algorithm for the ECS problem with single-news-article messages, two candidates, and arbitrary scores, then there is a ρ -approximation algorithm for DKS.*

Proof. Given an indirect graph G , we add a voter v_x for each vertex of X with preference rank $\langle 1, 0 \rangle$. For each edge n of N , we add a node v_n with preference rank $\langle 0, 2 \rangle$. Finally, for each edge $n = (x, x')$, we add an edge (with probability 1) from v_x to v_n and from $v_{x'}$ to v_n . The budget is B and it is easy to see that, in the optimal solution, the manipulator sends only messages $(1, 0)$ or $(0, -1)$.

First, we prove that for every solution S to the ECS problem, we can construct in polynomial time a solution with at least the same Δ_{MoV}^S and all the seeds in the set $\{v_x\}$. Let S be a solution to the ECS problem. Suppose S includes some nodes in v_n , where $n = (x, x')$. If both or neither of v_x and $v_{x'}$ are seeds, the seed in v_n is useless and can be removed. Otherwise, only v_x (or only $v_{x'}$) is a seed. In this case, we can replace the seed in v_n with a seed in $v_{x'}$ (or v_x) without decreasing Δ_{MoV}^S .

It can be observed that the value of Δ_{MoV}^S for a seed set S' that includes only seeds v_x is equal to $2|d(X')|$, where $X' = \{x \in X, v_x \in S'\}$ and $d(X')$ is the set of edges connecting nodes in X' . In particular, the set of edges that change the vote in favor of c_0 is the set of nodes v_n , where $n = (x, x')$ such that both v_x and $v_{x'}$ are seeds. This is exactly the number of edges in the subgraph X' . Since each new vote for c_0 decreases by one the votes for c_1 , it follows that $\Delta_{\text{MoV}}^S(S', M^*) = 2d(X')$.

Let A be an algorithm for a poly-time ρ -approximation algorithm for ECS that return a solution S' . Construct the solution S'' that includes only nodes in v_x . Let $X' = \{x \in X, v_x \in S''\}$ be the solution to DKS obtained from S'' . $\frac{d(X')}{d(X^*)} = \frac{\Delta_{\text{MoV}}^S(S'', M'')/2}{\Delta_{\text{MoV}}^S(S^*, M^*)/2} \geq \rho$. Thus from A , we can construct a polynomial time ρ -approximation algorithm for DKS.