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

Matteo Castiglioni
DEIB, Politecnico di Milano
Piazza Leonardo da Vinci, 32 – 20133, Milano (MI), Italy

Diodato Ferraioli
DIEM, Università degli Studi di Salerno
Via Giovanni Paolo II, 132 – 84084, Fisciano (SA), Italy

Nicola Gatti
Giuilia Landriani
DEIB, Politecnico di Milano
Piazza Leonardo da Vinci, 32 – 20133, Milano (MI), 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 theoretical 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 set of the seeds and the information sent. In particular, we study a wide range of cases distinguishing in the number of candidates or the kind of information spread over the network.

Our main result shows that the election manipulation problem is not affordable in the worst-case, even when one accepts to get an approximation of the optimal margin of victory, except for the case of seeding when the number of hard-to-manipulate voters is not too large, and the number of uncertain voters is not too small, where we say that a voter that does not vote for the manipulator’s candidate is hard-to-manipulate if there is no way to make her vote for this candidate, and uncertain otherwise.

We also provide some results showing the hardness of the problems in special cases. More precisely, in the case of seeding, we 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 (e.g., greedy, degree centrality, PageRank, VoteRank), fails to compute a bounded approximation even on elementary networks, such 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 election manipulation when the manipulator has an unlimited budget, being allowed to remove or add an arbitrary number of edges, and to the basic case of social influence maximization/minimization in the restricted case of finite budget.

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.

©2021 AI Access Foundation. All rights reserved.
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 have severe effects on our countries’ political life. Indeed, in many of the recent political elections worldwide, there has been evidence that false or incomplete news spread through these media influenced the electoral outcome. For example, in the 2016 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 (Allcott & Gentzkow, 2017; Guess et al., 2018). As another example, automated accounts on Twitter spread a considerable amount of political news to alter the 2017 French elections (Ferrara, 2017). 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 (Alaphilippe et al., 2018; Giglietto et al., 2018).

The increasing use of social networks to convey inaccurate and unverified information can lead to severe and undesired consequences, such as widespread panic, libelous campaigns, and conspiracies, representing 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.\footnote{We notice the election-manipulation problem is also studied under other perspectives, such as, e.g., Bayesian persuasion in offline (Castiglioni et al., 2020a; Castiglioni & Gatti, 2021) and online (Castiglioni et al., 2020b, 2021) settings.}

The former problem is known in the literature as election control through social influence, and it has recently been the subject of interest of many works in the artificial intelligence community. For instance, Sina et al. (2015) study a plurality voting scenario in which the voters can vote iteratively and show how to modify the relationship among voters to make the desired candidate win an election. Auletta et al. (2015, 2017a, 2017b) study a majority dynamics scenario and show that, in the case of only two candidates, a manipulator controlling the order in which information is disclosed to voters can lead the minority to become a majority. Auletta et al. (2018) study a similar adversary, showing that such a manipulator can lead a bare majority to consensus. These results do not extend to the case with more than two candidates, as showed by Auletta et al. (2019b, 2020). Bredereck and Elkind (2017) study a majority dynamics scenario, showing how selecting the seeds diffusing information to manipulate a two-candidate election. Finally, we mention a few works providing techniques to mitigate or block misinformation. For instance, Tsai et al. (2012) provide the seminal work on the application of approaches from the field of security games to social networks, and Auletta et al. (2020) study the problem of placing monitors to block malicious flaws in networks.

1.1 Related Work

Recently, Wilder and Vorobeychik (2018) 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 win or lose, respectively, the election. In particular, voters are not strategic, and ranks describe their preferences: given any pair of candidates $c, c'$ such that $c$ directly precedes $c'$ in the rank, positive information on $c$ or negative infor-
The diffusion of the information on the social network is described by the independent cascade model (Kempe et al., 2015). 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 voting rules and/or other diffusion models are adopted, as showed by Corò et al. (2019b, 2019a). 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 (Abouei Mehrizi et al., 2020).

The works by Wilder and Vorobeychik (2018) and Corò et al. (2019b, 2019a) present some limitations when dealing with elections with more than two candidates. A major limitation is the assumption that all the seeds send the same information, and this information is on a single candidate. Indeed, spreading simultaneously positive and/or negative information on multiple candidates is usually seen in most elections involving more than two candidates, in which manipulator not only send messages in favour of his/her candidate (or against its adversary), but it tries to support third parties so that they can erode votes from the main adversary. Not only, spreading 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 graph nodes) and five candidates $c_0, c_1, \ldots, c_4$, in which $c_0$ is the manipulator’s candidate.

---

2. To 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., (Becker et al., 2019).
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 and loses one position due to negative information on her. 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 the winner of the election (at most, the election ends with a tie with two votes taken both by $c_0$ and by either $c_1$ or $c_2$). Instead, injecting the network with positive 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 take to manipulate the election. However, this is not the case when the manipulator is (or collaborates 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 connected in the social network, or she may reveal information spread by unknown sources (e.g., as sponsored content or through friend suggestions mechanisms). That is, such a manipulator can remove or add edges in the network to obstruct or push the diffusion of information.\footnote{To the best of our knowledge, the removal or the addition of edges in the network have been studied as forms of manipulations only for simpler diffusion models, e.g., with two candidates and simple information diffusion dynamics (Bredereck & Elkind, 2017). They have been considered even when no information is spread, but voters update their votes in an iterative voting process by effect of selfish voting (Sina et al., 2015; Auletta et al., 2019).}

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 by Wilder and Vorobeychik (2018), along with two different directions. First, we address the problem that this model does not allow seeds to send messages on multiple candidates, as described above (cf., Example 1). For this reason, 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 use the term message to refer to the collection of information sent by every single seed. We extend the model introduced by Wilder and Vorobeychik (2018), that, in its turn, is an extension of the independent cascade model, to capture the simultaneous spread of multiple different messages.

The second direction along which our model differs from the previous one is that each message may have a different “effect” on the opinions of different voters, and on the opinions about different candidates by the same voter. That is, each voter keeps a ranking about candidates, but, differently from the model of Wilder and Vorobeychik (2018), 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. On the other side, our model also allows a candidate
to gain (lose, respectively) more than one position in the rank of a voter due to a large amount of received positive (negative, respectively) news articles about that candidate sent by a single seed or by multiple seeds. To this aim, we use the technical machinery to assign, for each voter, a value to each candidate. For each pair of candidates, the difference of their values represents how many positive (respectively, negative) news articles about the lowest (respectively, highest) ranked candidate are necessary to switch the relative order among them. Thus, a voter that is uncertain on $c_0$ can be modeled with the difference between the value assigned to the most preferred candidate and the value of $c_0$ being sufficiently small so that the manipulator can change the voter’s preferences to make $c_0$ be the most preferred with few messages. Conversely, a voter that is certain to vote for $c_0$ has a ranking such that $c_0$ is the most preferred candidate and values assigned to candidates such that the manipulator cannot make $c_0$ not be the most-preferred with few messages. Similarly, a voter that is certain not to vote for $c_0$ has a ranking such that $c_0$ is not the most preferred candidate and values such that the manipulator cannot make it be the most preferred. We say that such a voter is hard-to-manipulate.

In the paper, we often refer to a special basic setting, to which we refer as single-news-article messages, in which all the seeds send the same information, and this information is only on a single candidate. We also use the term unitary value distances to refer to the case in which the difference in the value of two candidates $c, c'$ where $c$ directly precedes $c'$ is exactly one. When our model is with single-news-article messages and unitary value distances and the number of candidates is two, it is directly comparable to that studied by Wilder and Vorobeychik (2018). Instead, with three or more candidates, the models are not comparable, as, differently from our model, a candidate cannot increase/decrease more than one position in the rank of a voter, even if the voter receives multiple positive/negative messages sent by multiple seeds (Wilder & Vorobeychik, 2018).

1.2.2 Approximation Bounds and Inapproximability 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 (2018). 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 number of news articles that can be spread over the network by seeds or the maximum amount of edges she can remove or add to the network. In Table 1, we summarize our main original results, providing a clear picture of the cases in which manipulation is affordable—and therefore countermeasures should be studied—and of the degree of inapproximability for all the other cases in which manipulation is not affordable. The main message is that the election manipulation problem is not affordable in the worst-case, even when one accepts to get an approximation of the optimal margin of victory, except for the case of seeding when the number of hard-to-manipulate voters is not too large with respect to the budget available to the manipulator, and the number of uncertain voters is not too small. From a prescriptive point of view, our result provides the manipulator with a lower bound on the budget to guarantee the election’s manipulability. On the opposite side, it provides the central authority with a criterion to make the election
Seeding

<table>
<thead>
<tr>
<th>Budget</th>
<th>Single-news-article Messages</th>
<th>General Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2 Candidates</td>
<td>2 or more Candidates</td>
</tr>
<tr>
<td></td>
<td>Unitary value distances</td>
<td>O((B)) hard-to-manipulate voters, (\delta) fixed</td>
</tr>
<tr>
<td></td>
<td>Arbitrary value distances</td>
<td>(\omega(B)) fraction of voters is uncertain</td>
</tr>
<tr>
<td>Limited</td>
<td>APX (Wilder &amp; Vorobeychik, 2018)</td>
<td>(\notin APX) (Thm 3)</td>
</tr>
<tr>
<td>Unlimited</td>
<td>(\nu) ((B)) hard-to-manipulate voters, (\omega(B)) fraction of voters is uncertain</td>
<td></td>
</tr>
</tbody>
</table>

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 value distances.

hard to manipulate. For instance, the central authority could pose an upper bound to the electoral campaigns’ costs motivated by, e.g., fairness arguments.

In the case of seeding, the problem is trivial when the budget \(B\) available to the manipulator is unlimited, as the manipulator can make all the nodes seeds spreading an arbitrarily large number of news articles in favor of \(c_0\) and against all other candidates. When instead the budget \(B\) is finite, our results depend on the scoring function of the 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 value distances become strictly larger than one. In the general setting (i.e., when messages and value distances are arbitrary), the complexity depends on the number of hard-to-manipulate voters, the number of uncertain voters, and on \(B\). We prove that, whenever the number of hard-to-manipulate voters is \(O(B)\) and there is at least a constant fraction of voters that is uncertain, there is a greedy polynomial-time algorithm guaranteeing an approximation factor \(\rho\) depending on \(\delta\) as \(\Omega(1/\delta)\), where \(\delta\) is the cost needed to make the most reluctant voter among those that are not hard-to-manipulate vote for \(c_0\). Observe that, in real-world elections, it is reasonable that, if a voter would vote for \(c_0\) when receiving an opportune set of news, then the cost \(\delta\) does not depend on the size of the network and is smaller than \(B\).

A surprisingly sharp transition phase occurs, instead, when these conditions are not satisfied. Indeed, we show that no polynomial-time approximation algorithm is possible, unless \(P = NP\), even when the approximation factor is a polynomial function of the size of the problem. 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, e.g., greedy, degree centrality, PageRank, VoteRank) fail to compute an empirically bounded approximation even on elementary networks, such as undirected graphs with every node
Election Manipulation on Social Networks

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 \( \text{NP} \)-hard even on graphs such 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. In the latter case, we study both the subcase with single-news-article messages and the more realistic subcase with arbitrary messages. In any of these cases, we show that deciding whether there is a set of edges to remove/add in the network to make the desired candidate win is hard, even when the value 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, to establish these results, we also provide new results for the basic Influence Optimization problem, which consists of maximizing or minimizing the number of nodes that receive the information spread over the network.\(^4\) We prove that the minimization (respectively, maximization) variant of the problem cannot be approximated within any constant factor by removing (respectively, adding) a limited number of edges.\(^5\)

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. Here, we present a seminal study in this direction, looking at manipulators that face a reoptimization problem, see e.g., (Ausiello et al., 2012, Chap. 4) and (Celli, Marchesi, & Gatti, 2017). We thus answer 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, where the social relationships among voters remain nearly stable between an election and the next one, or, in a single election, budget \( B \) could be available incrementally, and the manipulator could spend it in online fashion, thus reoptimizing every time that a new portion of the budget is available. 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.

\(^4\) Sheldon et al. (2010) studied the problem of adding edges to arbitrary nodes of the networks for maximizing the information diffusion, proving that this objective function is not submodular. Khalil et al. (2014) investigated these two types of graph modifications with the goal of minimizing the information diffusion. They show that this network structure modification problem has a supermodular objective. Heuristics for the edge removal problem have been studied by Kimura et al. (2008) and by Kuhlman et al. (2013). However, no hardness results are known.

\(^5\) For the sake of completeness, we mention that the Influence Optimization problem has been widely investigated when the manipulator uses seeding (Kempe et al., 2015).
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 4 provides our main results on the seeding problem. Section 5 provides results on edge removal, while Section 6 provides results on edge addition. Section 7 concludes the paper and describes future research directions.

2. Model and Problem Statement

We have a set of candidates $C = \{c_0, c_1, \ldots, c_\ell\}$ 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 directed edges, and $p: V \times V \rightarrow [0, 1]$ denotes the strength of the potential influence among voters. In particular, for each edge $(u, v)$, $p(u, v)$ returns the strength of the influence of $u$ on $v$.

Each voter $v$ has a ranking $r_v: C \rightarrow \{1, \ldots, |C|\}$ that models the preferences of voter $v$ over the candidates. Thus, $r_v(i) > r_v(j)$ models that voter $v$ prefers $c_i$ to $c_j$. We assume that $r_v$ is injective, thus returning a different ranking for each candidate, formally, $r_v(i) \neq r_v(j)$ $\forall c_i, c_j \in C$.

Each voter $v$ has also an injective value function $\pi_v: C \rightarrow \mathbb{R}$ which assigns a value to every candidate $c_i$. We assume that $\pi_v(i) > \pi_v(j)$ whenever $r_v(i) > r_v(j)$. Due to this property, we will often report only the value function of a voter, and not its ranking, since the latter can be immediately computed from the former. Values specify how easy it is that the voter changes her mind about the rank of voters. Specifically, for each pair of candidates $c_i, c_j$ such that $r_v(c_i) > r_v(c_j)$, $\pi_v(c_i) - \pi_v(c_j)$ is the cost necessary to convince $v$ to prefer $c_j$ to $c_i$. Sometimes we will denote by $\langle \pi_v(0), \ldots, \pi_v(\ell) \rangle$ the value vector of voter $v$. We will say that the values have unitary value distances if $\pi_v(i) \in \{0, \ldots, \ell - 1\}$ 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 rank in their preference ordering. For each candidate $c \in C$, we denote by $V_c$ the set of voters that rank $c$ first, formally, $V_c = \{v \in V \mid c = \arg\max_{c_i \in C} r_v(i)\}$.

Let $S \subseteq V$ be a subset of voters called seeds. Every seed $s$ can be selected to initiate the diffusion of information about multiple candidates. We denote by $m_s = (q_0, \ldots, q_\ell)$ 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 $c_i$. 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 \(v \notin S\), receiving a message from \(s\), accepts the information of this message 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 a live-graph as \(H = (V, E')\), 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 \(A^t_{m_s} \subseteq V\) composed of the active voters at time \(t\) due to message \(m_s\). Every set \(A^t_{m_s}\) is initialized with the seed sending the corresponding message for \(t = 0\), i.e., \(A^0_{m_s} = \{s\}\), and the empty set for \(t > 0\). At every time \(t \geq 1\), set \(A^t_{m_s}\) is defined as follows: for every edge \((u, v) \in E'\), we consider the set \(M(u, v) \subseteq M\) of messages \(m_s\) such that \(u \in A^{t-1}_{m_s}\)—and thus \(u\) has just been activated by \(m_s\)—and \(v \notin \bigcup_{i < t} A^i_{m_s}\)—and thus \(v\) has never been activated by \(m_s\); then for each \((u, v)\) such that \(M(u, v)\) is not empty, we add \(v\) to \(A^t_{m_s}\) for every \(m_s \in M(u, v)\). The diffusion process of message \(m_s\) terminates at time \(T_{m_s}\) when \(A^{T_{m_s}}_{m_s} = \emptyset\). Finally, the cascade terminates when the diffusion of every message \(m_s\) terminates. A voter that activates at some \(t\) is said to be influenced. Note that, when the messages are single-news-article, there are two candidates, and the value distances are unitary, this process reduces to the well-known independent cascade model (Kempe et al., 2015).

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 \(\phi\) associates each pair \((\pi, R)\) with a new ranking \(r'\). The ranking revision function updates first the values \(\pi^i\) such that each positive (negative) message \(m_s\) on a candidate \(c_i\) increases (reduces) \(\pi_i\) by \(m_s(i)\). In principle, for each voter \(v\) and each candidate \(c_i\), we aim at updating the value of each candidate as follows:

\[
\pi'_v(i) \leftarrow \pi_v(i) + \sum_{m_s \in R} m_s(i),
\]

and subsequently we set the new ranking \(r'_v\) such that \(r'_v(i) > r'_v(j)\) only if \(\pi'_v(i) > \pi'_v(j)\). However, such a procedure can return a non-strict ranking at the end of the diffusion process in which multiple candidates have the same value of \(\pi_v\). In order to guarantee that the ranking is strict, we break ties according to some rule and slightly tilt values so that they satisfy the tie-breaking outcome. For the sake of simplicity, given two candidates with the same \(\pi'_v\), we break ties in favor of the candidate ranked last before the diffusion process, i.e., that with the smaller \(\pi_v\). A simple interpretation is that we assume that voters will give more weight to “fresher” received news than to their belief, usually built on “old” news. Such a tie-breaking rule can be obtained by slightly perturbing the initial
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' values for candidates, from which their ranking can be immediately computed.

value with a multiplicative factor $(1 - \epsilon)$, where $\epsilon$ is a sufficiently small positive constant, e.g., $\epsilon = \frac{1}{1 + \max_{v \in V} \pi_v(i)}$, and then apply the update rule as

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

Given a seed set $S$, a set $M$ of messages, a set $E$ of edges, and a live graph $H$, $r^*_{c}(i, S, M, E, H)$ denotes the rank that voter $v \in V$ assigns to 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 first after the preference revision, i.e., $V_c^*(S, M, E, H) = \{v \mid \arg\max_{c_i} r^*_c(i, S, M, E, H) = c\}$. Finally, we define the margin of victory MoV of $(S, M, E, H)$ as

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

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 value 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 value to $(1 - \epsilon, 2 - 2\epsilon)$ 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 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 values to $\langle 2, 2 - \epsilon, 2 - 2\epsilon \rangle$ and then voting for $c_0$. Hence, $\text{MoV}(S, M, E, H_2) = 3 - \max\{1, 1\} = 2$.

### 2.3 Election Control Problem

The election control problem involves a 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 and network modification by edge removal. 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 the election scenario $(C, G, \{\pi_v\})$ and budget $B \in \mathbb{N}$, the goal is to find a set $S$ of seeds and messages $m_s$ for every $s \in S$, with $|M| \leq B$, to maximize $\mathbb{E}_H[\Delta^\text{S}_{\text{MoV}}(S, M, H)]$, where $\Delta^\text{S}_{\text{MoV}}(S, M, H) = \text{MoV}(S, M, E, H) - \text{MoV}(\emptyset, \emptyset, E, H)$ is the increase of $\text{MoV}$ due to the messages send by the seeds $S$.

**Definition 2 (Election-Control-by-Edge-Removal (ECER)).** Given the election scenario $(C, G, \{\pi_v\}, S, M)$ and budget $B \in \mathbb{N} \cup \{\infty\}$, the goal is to find $E' \subseteq E$ 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 $\text{MoV}$ due to the removal of edges $E'$.

**Definition 3 (Election-Control-by-Edge-Addition (ECEA)).** Given the election scenario $(C, G, \{\pi_v\}, S, M)$ and budget $B \in \mathbb{N} \cup \{\infty\}$, the goal is to find $E'$ 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 $\text{MoV}$ due to the addition of edges $E'$.

An algorithm $A$ is said to always return a $\rho$-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 pair $(S, M)$ such that $\mathbb{E}_H[\Delta_{\text{MoV}}(S, M, H)] \leq \rho \mathbb{E}_H[\Delta_{\text{MoV}}(S^*, M^*, H)]$, where $(S^*, M^*)$ is the optimal solution of the problem. A similar definition holds for all the other optimization problems.\(^6\)

---

\(^6\) Since our results are mainly hardness results, it is immediate to see that they extend to more complex (but more realistic) settings: e.g., we can allow the same message to cause a different value increment (decrement) to different voters, or if received by different neighbors, or if sent by different seeds. Also they continue to hold if the diffusion probabilities depend on the content of the message (e.g., positive messages about the less preferred candidate are transmitted with lower probability). Finally, as highlighted later in the paper, our hardness results can be extended to hold even for additive approximation, and not only for multiplicative approximation.
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 to find 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 to find 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 value distances are arbitrary. Before stating our characterization, we need to introduce some parameters. First, for every \( v \in V \setminus V_{c_0} \), we denote by \( \delta_v \) the cost the manipulator needs to spend to convince \( v \) to vote for \( c_0 \) and we set \( \delta_v = \max_{c_i \neq c_0} \{\pi_v(i) - \pi_v(0)\} \). Given an ECS problem instance, a voter \( v \in V \setminus V_{c_0} \) is called hard to manipulate if \( B < \delta_v \), and uncertain otherwise. Let \( V^H \) and \( U \) be the set of voters that in the given ECS problem instance are hard to manipulate and uncertain, respectively.

Our characterization is based on the parameter

\[
\delta = \max_{v \in V \setminus V^H} \delta_v,
\]

representing the cost the manipulator needs to spend to convince the most reluctant voter that is not hard to manipulate to vote for \( c_0 \).

3.1 Approximation Results

We next show a polynomial-time approximation algorithm for instances in which the number of hard-to-manipulate voters is not too large with respect to the budget \( B \), and the fraction of uncertain voters is not too small.

The following lemma describes the algorithm and bounds its approximation on general instances. We next show that this provides an \( \Omega(1/\delta) \) approximation whenever the number of hard-to-manipulate voters is \( O(B) \) and there is at least a constant fraction of voters that are uncertain.

---

7. It may appear strange that a problem of influence minimization has an objective function to be maximized. However, our definition is actually motivated by the choice of keeping the objective function non-negative.
Lemma 1. For every $\varepsilon > 0$, there is a greedy polynomial-time algorithm returning a solution $(\tilde{S}, \tilde{M})$ with

$$
\mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)] \leq 2B \lambda \left( 1 - \frac{1}{e} - \varepsilon \right)^{-1} \mathbb{E}_H[\Delta_{\text{MoV}}^S(\tilde{S}, \tilde{M}, H)] + |V^H|,
$$

where $(S^*, M^*)$ is the optimal solution to the ECS problem, and $\lambda = \left\lfloor \frac{B}{\varepsilon} \right\rfloor$.

Moreover, if $0 < |U| < \lambda$, then $(\tilde{S}, \tilde{M})$ is a $\rho$-approximation of $(S^*, M^*)$ for $\rho = \frac{2|U|}{\lambda}$.

Proof. Let $\hat{m}$ with $|\hat{m}| = \delta$ be the message that causes each non-hard-to-manipulate voter to vote for $c_0$, whatever was the ranking before the reception of this message. E.g., this message may have $\hat{m}(0) = \delta$ and $\hat{m}(i) = 0$ for $i \neq c_0$. Consider the function $\tilde{\chi}(S, H)$ returning the number of influenced voters that are not hard to manipulate and do not prefer $c_0$ before the reception of the message. Suppose we are given a procedure $A$ that, on input $b$, returns in polynomial time a set $\tilde{S}$ of $b$ seeds such that $\mathbb{E}_H[\tilde{\chi}(\tilde{S}, H)] \geq r \max_{\ell} \mathbb{E}_H[\tilde{\chi}(\tilde{S}, H)]$, for some $r \leq 1$. Our algorithm then selects $\lambda = \left\lfloor \frac{B}{\varepsilon} \right\rfloor$ seeds through procedure $A$, and let each of these seeds send the message $\hat{m}$.

In order to formally prove the approximation factor of this algorithm, recall that $\tilde{S}$ is the set of seeds returned by procedure $A$ on input $\lambda$. Moreover, we denote with $(S^*, M^*)$ the set of seeds and messages maximizing $\mathbb{E}_H[\Delta_{\text{MoV}}^S(S, M, H)]$ i.e., the optimal solution of the manipulation problem. We also denote with $S'$ ($S''$, respectively) the set of seeds of size $B$ ($\lambda$, respectively) that maximizes $\mathbb{E}_H[\tilde{\chi}(S, H)]$. Note that, by our definition of procedure $A$, we have that $\mathbb{E}_H[\tilde{\chi}(\tilde{S}, H)] \geq r \mathbb{E}_H[\tilde{\chi}(S'', H)]$.

Specifically, there exists an implementation of procedure $A$ such that $r = 1 - \frac{1}{e} - \varepsilon$ for $\varepsilon > 0$. Indeed, it is known that the function $\mathbb{E}_H[\tilde{\chi}(S, H)]$ is monotone and submodular on $S$ (Kempe et al., 2015), i.e., $\mathbb{E}_H[\tilde{\chi}(S, H)] \leq \mathbb{E}_H[\tilde{\chi}(T, H)]$ and $\mathbb{E}_H[\tilde{\chi}(S \cup \{x\}, H)] - \mathbb{E}_H[\tilde{\chi}(S, H)] \geq 0$ for every $S \subseteq T$ and every $x \notin T$.

Consequently, the greedy algorithm, that selects at each time the seed that increases the desired quantity the most, 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 (Kempe et al., 2015). Unfortunately, such a naive greedy algorithm does not work in polynomial time, since the function $\mathbb{E}_H[\tilde{\chi}(S, H)]$ cannot be evaluated within this time limit. However, there is a fully polynomial approximation scheme returning an $\varepsilon$-approximation to $\mathbb{E}_H[\tilde{\chi}(S', H)]$ for each $\varepsilon > 0$. Adopting this approximation in place of the real value of $\mathbb{E}_H[\tilde{\chi}(S, H)]$ in the greedy algorithm may cause the approximation ratio increase by an additive factor of $\varepsilon$, as showed by Kempe et al. (2015). Hence, we have that:

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

Note that $|V_c| - \mathbb{E}_H[|V_c^*(S^*, M^*, H)|] \leq \mathbb{E}_H[\tilde{\chi}(S^*, H)]$ 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

$$
\max_{c \neq c_0} |V_c| - \mathbb{E}_H[\max_{c \neq c_0} |V_c^*(S^*, M^*, H)|] \leq \max_{c \neq c_0} \left\{ |V_c| - \mathbb{E}_H[|V_c^*(S^*, M^*, H)|] \right\}
$$

$$
\leq \mathbb{E}_H[\tilde{\chi}(S^*, H)] + |V^H|.
$$

1061
A similar argument proves that
\[ \mathbb{E}_H \left[ |V_{c_0}^*(S^*, M^*, H)| \right] - |V_{c_0}| \leq \mathbb{E}_H \left[ \hat{\chi}(S^*, H) \right]. \] (3)

Moreover, by submodularity of \( \hat{\chi} \), it holds that \( \mathbb{E}_H[\hat{\chi}(S', H)] \leq \frac{\mathbb{E}_H[\hat{\chi}(S'', H)]}{|S'|} \). Hence, since \( |S'| = B \) and \( |S''| = \lambda \), we achieve that
\[ \mathbb{E}_H \left[ \hat{\chi}(S', H) \right] \leq \frac{B}{\lambda} \mathbb{E}_H \left[ \hat{\chi}(S'', H) \right]. \] (4)

Moreover, by the definition of \( \Delta_{\text{MoV}}^S \) (see Definition 1), we have that
\[ \mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)] = \mathbb{E}_H \left[ |V_{c_0}^*(S^*, M^*, H)| \right] - \mathbb{E}_H \left[ \max_{c \neq c_0} |V_c^*(S^*, M^*, H)| \right] - \left( |V_{c_0}| - \max_{c \neq c_0} |V_c| \right). \]

Hence, we directly achieve that
\[ \mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)] = \left( \mathbb{E}_H \left[ |V_{c_0}^*(S^*, M^*, H)| \right] - |V_{c_0}| \right) + \left( \max_{c \neq c_0} |V_c| - \mathbb{E}_H[\max_{c \neq c_0} |V_c^*(S^*, M^*, H)|] \right). \]

Then, have that
\[ \mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)] \leq 2 \mathbb{E}_H \left[ \hat{\chi}(S^*, H) + |V^H| \right] \] (by (2) and (3))
\[ \leq 2 \mathbb{E}_H \left[ \hat{\chi}(S', H) + |V^H| \right] \] (by definition of \( S' \))
\[ \leq \frac{2B}{\lambda} \mathbb{E}_H \left[ \hat{\chi}(S'', H) + |V^H| \right] \] (by (4))
\[ \leq \frac{2B}{\lambda} \left( 1 - \frac{1}{e} - \epsilon \right)^{-1} \mathbb{E}_H \left[ \hat{\chi}(S', H) + |V^H| \right] \] (by (1))
\[ \leq \frac{2B}{\lambda} \left( 1 - \frac{1}{e} - \epsilon \right)^{-1} \mathbb{E}_H \left[ \Delta_{\text{MoV}}^S(S', \hat{M}, H) + |V^H| \right], \]

where the last inequality follows from the fact that, by definition of \( \hat{n} \), all the influenced nodes will vote for \( c_0 \).

As for the case that \( 0 < |U| < \lambda \), observe that the maximum number of voters that may be influenced to vote for \( c_0 \) is \(|U|\), and these voters are all selected by the algorithm above. Hence, \( \mathbb{E}_H \left[ \Delta_{\text{MoV}}^S(S', \hat{M}, H) \right] \geq |U| \), whereas it must be the case that \( \mathbb{E}_H[\Delta_{\text{MoV}}^S(S^*, M^*, H)] \leq 2|U| + |V^H| \), from which the desired approximation ratio follows.

Hence, we have the following corollary.

**Corollary 1.** There is a greedy polynomial-time algorithm returning an \( \Omega \left( \frac{1}{\Delta} \right) \)-approximation to the ECS problem, whenever (i) the uncertain voters are at least a constant fraction of the voters that do not vote for \( c_0 \), i.e., \(|U| = \Omega(|V \setminus V_{c_0}|)\); (ii) the hard-to-manipulate voters are not too many with respect to the budget \( B \), i.e., \(|V^H| = O(B)\).
Proof. Consider first the case that $|U| = |V \setminus V_{c_0}| - |V^H| < \lambda$. Then, according to Lemma 1, we have that there is a $\rho$-approximation greedy algorithm with

$$\rho = \frac{|U|}{2|U| + |V^H|} = \frac{|U|}{|U| + |V \setminus V_{c_0}|} = \Omega(1) = \Omega \left( \frac{1}{\delta} \right),$$

where the last equality follows since $\delta \geq 1$.

Now we consider the case that $|U| > \lambda$. Observe that in this case $\mathbb{E}_H \left[ \Delta_{MoV}^S(\hat{S}, \hat{M}, H) \right] \geq \lambda$, since at least the selected seeds are influenced to vote for $c_0$.

From Lemma 1, we have that a greedy polynomial-time algorithm exists returning a solution $(\hat{S}, \hat{M})$ to the ECS problem with

$$\mathbb{E}_H[\Delta_{MoV}^S(S^*, M^*, H)] \leq 2B \left( \frac{1}{\lambda} - \frac{1}{e} - \epsilon \right)^{-1} \mathbb{E}_H \left[ \Delta_{MoV}^S(\hat{S}, \hat{M}, H) \right] + |V^H|,$$

where $(S^*, M^*)$ is the optimal solution to the ECS problem, and $\lambda = \lfloor \frac{B}{2} \rfloor$.

If $|V^H| = 0$, then we achieve that the solution returned by the algorithm is a $\rho$-approximation of the optimal solution for $\rho = \frac{1}{2} \left( 1 - \frac{1}{e} - \epsilon \right) \frac{\lambda}{B} = \Theta \left( \frac{\lambda}{B} \right)$. If, instead, $|V^H| > 0$, then we have that

$$\mathbb{E}_H[\Delta_{MoV}^S(S^*, M^*, H)] \leq |V^H| \left( \frac{2}{|V^H| \lambda} \left( 1 - \frac{1}{e} - \epsilon \right) \mathbb{E}_H \left[ \Delta_{MoV}^S(\hat{S}, \hat{M}, H) \right] + 1 \right) \leq \left( \frac{2}{|V^H| \lambda} \left( 1 - \frac{1}{e} - \epsilon \right) \mathbb{E}_H \left[ \Delta_{MoV}^S(\hat{S}, \hat{M}, H) \right] + 1 \right).$$

Observe that $\left( 1 - \frac{1}{e} - \epsilon \right)^{-1} \geq 1, \frac{\mathbb{E}_H[\Delta_{MoV}^S(\hat{S}, \hat{M}, H)]}{\lambda} \geq 1$, and there is a constant $c$ such that $\frac{|V^H|}{|V^H|} \geq \frac{1}{c}$. Then, the second term within parenthesis in Equation 5 is at most $\frac{c}{2}$ times larger than the first term, and we then conclude that

$$\mathbb{E}_H[\Delta_{MoV}^S(S^*, M^*, H)] \leq (c + 2) \frac{B}{\lambda} \left( 1 - \frac{1}{e} - \epsilon \right) \mathbb{E}_H \left[ \Delta_{MoV}^S(\hat{S}, \hat{M}, H) \right].$$

Then the algorithm returns a solution that is a $\rho$-approximation of the optimal solution for $\rho = \frac{1}{c+2} \left( 1 - \frac{1}{e} - \epsilon \right) \frac{\lambda}{B} = \Theta \left( \frac{\lambda}{B} \right)$.

To conclude the proof, we observe that $\frac{\lambda}{B} > \frac{1}{25}$, giving in this way the desired approximation. Indeed, if $\lambda = 1$, then $\delta > \frac{2}{B}$ and thus $\frac{\lambda}{B} = \frac{1}{B} > \frac{1}{25}$; otherwise $\lambda \geq \frac{B - \delta + 1}{\delta}$ and $\delta \leq \frac{B}{2}$, from which, we have that $\frac{\lambda}{B} \geq \frac{B/2 + 1}{B} \delta > \frac{1}{25}$. \hfill \Box

Corollary 1 essentially proves that in case the number of hard-to-manipulate voters is not too large with respect to $B$ and the fraction of uncertain voters is at least constant, the problem of election manipulation essentially reduces to the problem of influence maximization. Hence, it should be expected that not only the theoretically results, but also experimental results known for the latter, such as the good performance of different heuristics, will extend to election control.

Our algorithm heavily depends on the possibility that the seeds can send messages with information on multiple candidates. As we will see later in Section 3.2, Theorem 3 shows that, without this possibility, it is unlikely that the ECS problem is approximable within a constant factor even with two candidates.
3.2 Inapproximability Results

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

**Definition 6 (Set-Cover).** Given a set \( N = \{z_1, \ldots, z_n\} \) of \( n \) elements, a collection \( X = \{x_1, \ldots, x_g\} \) of sets with \( x_i \subset N \), and a positive integer \( h < g \), the objective is to select a collection \( X^* \subset X \) with \( |X^*| \leq h \) and \( \bigcup_{x_i \in X^*} x_i = N \).

**Theorem 1.** For any \( \rho > 0 \) even depending on the size of the problem, there is no algorithm that on input instances of the ECS problem with at least three candidates, always returns a \( \rho \)-approximation, 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.\(^8\) 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: \( \pi_v = (0,B + 1,B) \).

The component \( G_3 \) is a clique of \( n + 2g - h + 3 \) nodes, such that \( n + g + 2 \) nodes have a preference \( \pi_v = (B,B - 1,B - 2) \) and \( g - h + 1 \) nodes with \( \pi_v = (0,B + 1,0) \).

\(^8\) If \( |C| > 3 \), we can focus on instances in which all candidates except three of them are dummy. To this aim, it is sufficient to set \( \pi_v(i) = \max\{\pi_v(0),\pi_v(1),\pi_v(2)\} - B - 1, \forall i \in \{3, \ldots, C - 1\}, v \in V \). Hence, there is no way for a manipulator having budget \( B \) to increase the ranking of these candidates to the first position. Hence they cannot influence the margin of victory of the manipulator’s candidate.
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 \(|V_c| = g+n+2, |V_{c1}| = g+n+2, \) and \(|V_{c2}| = g+n\). Hence, MoV(\( \emptyset, \emptyset, E, H \)) = 0.

We next prove that this instance has a solution \( (S^*, M^*) \) with 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 \( \bigcup_{x_i \in X^*} = N \)).\(^9\) 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 \( m_{v_0}^* \) 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 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, \(|V_{c1}^*(S^*, M^*, H)| = |V_{c1}| - |G_2| + h + n = g + n + 2 - n - h - 1 + n + h = g + n + 1, \) and \(|V_{c2}^*(S^*, M^*, H)| = |V_{c2}| + |G_3| - h - n = g + n + 1 \). Hence, MoV\( (S^*, M^*, E, H) = 1, \) as desired.

**Only if.** Suppose that there exists a pair \( (S^*, M^*) \) such that MoV\( (S^*, M^*, E, H) > 0 \). Note that, since \( \delta > B \), \( c_0 \) cannot gain votes and \(|V_{c0}^*(S^*, M^*, H)| = |V_{c0}| \). Hence, in order to have 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 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 \(|V_{c2}^*(S^*, M^*, E, H)| > g + n + n + h + 1 - (n + h) \) and thus 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 change their vote from \( c_2 \) to \( c_1 \). Observe that, in order to make a voter change, a single message is sufficient (i.e., 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 incoming 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 solution \( (S^*, M^*) \) satisfying \( \Delta_{\text{MoV}}(S^*, M^*, H) > 0 \) exists if and only if a solution for the Set-Cover instance exists. Note also that if a solution \( \Delta_{\text{MoV}}(S^*, M^*, H) > 0 \) exists, then \( \Delta_{\text{MoV}}(S, M, H) > 0 \) even for any \( \rho \)-approximate solution \( (S, M) \), regardless of the value of \( \rho \). Thus, if a polynomial-time \( \rho \)-approximation

---

9. If there is a solution of Set-Cover \( X^* \) of size less than \( h \), then we can achieve a solution of Set-Cover of size exactly \( h \), by padding \( X^* \) with arbitrary elements in \( X \setminus X^* \).
algorithm for election control problem exists, then the Set-Cover problem can also be solved in polynomial time, implying that $P = NP$.\footnote{This reduction can be used to provide additive hardness replacing each node with a large clique: larger it is the value of the clique, larger it will be the gap among solutions corresponding to a yes and no instances of Set-Cover, and this larger will be the approximation ratio. A similar trick will prove hardness of additive approximation for each of the following reductions.}

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 are no uncertain voters. Moreover, it is not hard to check that the argument of Theorem 1 can be used to prove that no approximation algorithm is possible even if one requires that the algorithm runs only in fixed-parameter-tractable time with respect to the budget, unless $FPT = W[2]$.

The instance given in the proof of Theorem 1 is very extreme, since it does not present any uncertain voter. We next show that, even by dropping this assumption, we still achieve inapproximability results. Specifically, we show in Proposition 1 that no constant approximation is achievable in polynomial time even if the number of uncertain voters is non-zero, but $o(|V \setminus V_{c_0}|)$. Similarly, in Proposition 2, we show that no constant approximation is achievable in polynomial time, even if the number of uncertain voters is $\Omega(|V \setminus V_{c_0}|)$, whenever the number of hard-to-manipulate voters is $\omega(B)$. Hence, Proposition 1 and Proposition 2 show that conditions of Corollary 1 are essentially tight.

**Proposition 1.** For any $\rho > 0$, there is no algorithm that, on input instances of the ECS problem with at least three candidates and at most $|U| = o(|V \setminus V_{c_0}|)$ uncertain voters, always returns a $\rho$-approximate solution, unless $P = NP$.

*Proof.* The reduction is similar to the one described in the proof of Theorem 1. Specifically, let us consider the instance built therein, we modify this instance so that there are $u$ uncertain voters, with $u$ that will turn out to be $o(|V \setminus V_{c_0}|)$. Specifically, we apply the following changes, with $\mu = \nu = u$:

- each node in $G_1$, $G_2$, and $G_3$ is enlarged into a clique of size $\mu \rho'$, where $\rho' > \rho$;
- remove $\nu$ nodes from the $\mu \rho'(g - h + 1)$ nodes with preference ranking $\pi = (0, B + 1, 0)$ in $G_3$;
- add $\nu$ isolated nodes with preference ranking $\pi = (B, B + 1, 0)$.

Hence, there are $\nu$ uncertain voters, and $2\mu \rho'(n + g + 1) - \nu = \omega(u)$ hard-to-manipulate voters.

Now, if a set cover of size at most $h$ exists, then, by mimicking the arguments in the proof of Theorem 1, we have that $\Delta_{MoV}(S^*, I^*, H) \geq u \rho'$. Otherwise, there is no way of incrementing $\Delta_{MoV}$ by intervening on $G_1$, $G_2$, and $G_3$, and hence the only nodes that eventually change opinion are the $u$ uncertain nodes. Thus, any $\rho$-approximation algorithm must be able to distinguish between these two cases and thus solves the Set-Cover problem in polynomial time. \qed
Proposition 2. For any \( \rho > 0 \), there is no algorithm that on input instances of the ECS problem with at least three candidates and at least \( |V^H| = \omega(B) \) hard-to-manipulate voters, always returns a \( \rho \)-approximate solution, unless \( P = NP \).

Proof. This reduction is exactly the same as the previous one, except that now we set \( \mu = h + 1 \) and \( \nu = \mu \rho'(g - h + 1) \), and we may add up to \( n + g \) uncertain isolated voters for each remaining candidate \( c_i \), if any, with preference ranking \( \pi(0) = B, \pi(i) = B + 1 \) and \( \pi(j) = 0 \) for every remaining candidate. Hence, there are \( (h + 1)\rho'(n + g + h + 1) = \omega(B) \) hard-to-manipulate voters.

As above, if a set cover of size at most \( h \) exists, then \( \Delta_{\text{MoV}}(S^*, I^*, H) \geq (h + 1) \rho' \). Otherwise, there is no way of incrementing \( \Delta_{\text{MoV}} \) by intervening on \( G_1, G_2, \) and \( G_3 \), and hence it is only possible to change the opinion of those uncertain nodes that are selected as seeds, that are at most \( h + 1 \). Thus any \( \rho \)-approximation algorithm must be able to distinguish between these two cases and thus solves the Set-Cover problem in polynomial time.

3.2.1 Further Hardness Results

Theorem 1 (and Propositions 1 and 2) does not rule out that the worst-case instances are very rare and/or knife-edge. However, we show that simple 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 degree two or trees.

Specifically, given a set \( S \) of seeds and corresponding messages \( M \), we denote as \( F(S, M) \) the set of pairs \( (s, m_s) \), with \( s \notin S \) such that either

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

or

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

That is, \( F(S, M) \) 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 \( M = \emptyset \); until the set \( F(S, M) \) is not empty, choose one \( (s, m_s) \in F(S, M) \), set \( S = S \cup \{s\} \) and \( M = M \cup \{m_s\} \), and update \( F(S, M) \) accordingly. We show that every algorithm following the greedy approach fails even for elementary networks.

Proposition 3. For any \( \rho > 0 \) even depending on the size of the problem, there is no \( \rho \)-approximation algorithm following the greedy approach for the ECS problem, even in undirected graphs in which each node has degree at most 2.

Proof. Consider the graph given in Figure 4: 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 4: Example of a small undirected network in which the greedy algorithm performs badly.

Since, for all nodes of $G_2$ and $G_3$, the value 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 $\mathcal{F}(\emptyset, \emptyset)$ is empty, and thus the algorithm never adds any seed to $S$:

- clearly, $\mathcal{F}(\emptyset, \emptyset)$ 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 $\rho$-approximation.\footnote{11. It is not hard to check that on this simple instance a failure in looking for the unique solution providing positive increment of margin of victory occurs even if we use common centrality measures (that are often...}
A similar result holds even by considering directed trees.

**Proposition 4.** For any $\rho > \frac{38}{|V|}$, there is no $\rho$-approximation algorithm following the greedy approach for the ECS problem, even in directed trees.

**Proof.** Consider the graph in Figure 5. It is composed of 5 subgraphs $A_1, A_2, A_3, A_4, A_5$, where $A_1$ is a directed line of $7r$ 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 $\rho > \frac{2}{2r}$. 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 $B = 2$.

![Figure 5: Example of a tree in which the greedy algorithm performs badly.](image)

Due to the budget constraint, $c_0$ cannot get any further vote, and thus, to increase the margin of victory of $c_0$, $c_2$ must get some of the $c_1$’s votes. The optimal solution $(S^*, M^*)$ is then obtained by setting that $S^* = \{x, y\}$ (see in Figure 5 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 adopted as successful heuristics for the influence maximization problem). For instance, degree centrality and many other measures based on it, such as voteRank, will never select a node from $G_{3,2}$ as a seed, regardless of the adopted tie-breaking rule; instead PageRank, and all path-based centrality measure will select a seed within $G_1$. 

1069
victory is \( r \). However, this solution cannot be found by any algorithm adopting the greedy approach. Indeed, we have that no pair \((s, m_a)\) 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 change its vote from \( c_1 \) to \( c_2 \). By repeating the argument, we have that the 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 \( \frac{2}{r} < \rho \).

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 (Kempe et al., 2015). 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, by showing that maximizing the expected \( \Delta_{\text{MoV}} \) is \( \text{NP} \)-hard even on a line. Recall that, while Influence Maximization by Seeding is \( \text{NP} \)-hard for arbitrary graphs, there exists a polynomial-time algorithm when the graph is a line (Wang et al., 2016).

**Theorem 2.** The ECS problem with at least four candidates is \( \text{NP} \)-hard even on line graphs.

**Proof.** We reduce from \textsc{Partition}. This problem, given a set of positive integers \( A = \{a_1, a_2, ..., a_n\} \), 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 \( \text{NP} \)-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 \textsc{Partition}, we build a graph as in Figure 6. 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.\(^{12}\) There are \( 5n \) isolated nodes with preference rank \( (0, B + 2, B + 1, B) \), \( 8n + 1 - 8k \) isolated nodes with preference rank \( (0, B + 1, B + 2, B) \) and \( 3n \) nodes with preference rank \( (4, 3, 2, 1) \). Each line is composed by 5 nodes. For each line \( i \in \{1, 2, ..., n\} \), the first three nodes have preference rank \( (0, B + 2, B + 1, 1) \), the last two nodes have preference rank \( (0, B, B + 1, B + 2) \), 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, respectively, with probability \( 1 - p_i \) and \( w_i \), where:

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

Note that, since \( a_i < t, \forall i \in \{1, ..., n\} \), then \( p_i < \frac{1}{4}, \forall i \). It is easy to see that, since \( p_i < \frac{1}{4}, \forall i \), 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, \( \text{MoV}(\emptyset, \emptyset, E, H) = 0 \).

\(^{12}\) The graph can be seen as a single line with zero probability links going from \( v_{e,j} \) to \( v_{a,j+1} \), for \( j = 1, 2, ..., n - 1 \), and connecting the isolated nodes.
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 \(8n + 1 - 3k\) votes. When at least an edge is not active, the margin of victory is given by the number of nodes lost by \(c_1\). Hence, the expected margin of victory is

\[
\mathbb{E}_H[\Delta_{\text{MOV}}^S(S', M, H)] = 2k + \sum_{i \in K'}(1 - p_i) - \prod_{i \in K'}(1 - p_i)w_i
\]

\[
= 3k - \sum_{i \in K'}p_i - \prod_{i \in K'}(1 - p_i)w_i.
\]
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 by 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 $S'' = \{v_{a,i} | i \in R(k)\}$. $E_H[\Delta_{\text{MoV}}(S'', M, H)] = 3k - \sum_{i \in R(k)} p_i - \prod_{i \in R(k)} (1 - p_i) w_i > 3k - 1 - \sum_{i \in R(k)} p_i$, where the last inequality follows from $\prod_{i \in R(k)} (1 - p_i) w_i < 1$. It follows that $E_H[\Delta_{\text{MoV}}^*(S^*, M, H)] \leq |S^*_1|(2 - p_i) + \sum_{i \in R(|S^*| - |S^*_1|)} (3 - p_i) < E_H[\Delta_{\text{MoV}}^*(S'', M, H)]$.

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 $E_H[\Delta_{\text{MoV}}^*(S^*, M^*, H)]$ 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 $P = NP$.

We conclude the section by 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 value distances are two. We reduce from DENSEST-K-SUBGRAPH whose definition follows.

**Definition 7 (DENSEST-K-SUBGRAPH (DkS)).** Given an undirected graph $G = (X, N)$, find the set $X^*$ of $k$ vertices that maximizes $d(X^*) = |E'|$, where $G(X^*) = (X^*, E')$ is the subgraph of $G$ with vertices $X^*$.

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

**Proof.** Given an instance $(G, k)$ for DkS, with $G$ being an undirected graph and $k$ being an integer, we build an instance $I(G, k)$ for the ECS problem as follows. Given an undirected graph $G$, we add a voter $v_x$ for each vertex of $X$ with preference rank $(1, 0)$. For each edge $n$ of $N$, we add a node $v_n$ with preference rank $(0, 2)$. 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 $\Delta_{\text{MoV}}^S$ 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 $\Delta_{\text{MoV}}^S$.

It can be observed that the value of $\Delta_{\text{MoV}}^S$ for a seed set $S'$ that includes only seeds $v_x$ is equal to $2d(X')$, 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 $\Delta_{\text{MoV}}^S(S', M^*) = 2d(X')$.  

1072
Let $A$ be a polynomial-time $\rho$-approximation algorithm for the ECS problem, and let $S'$ be the solution returned by $A$ on the instance $I(G,k)$. Construct the solution $S''$ that includes only nodes in $v_x$ using the procedure described above. Let $X' = \{x \in X,v_x \in S''\}$ be the solution to $DkS$ obtained from $S''$. Thus, we have that:

$$d(X') = \frac{\Delta^{S}_{MoV}(S'',M'')/2}{\Delta^{S}_{MoV}(S^*,M^*)/2} \geq \rho.$$ 

Therefore, from $A$, we can construct a polynomial-time $\rho$-approximation algorithm for $DkS$. 

Manurangsi (2017) shows that there is no constant-factor polynomial time approximation algorithm for the $DkS$ problem unless the Exponential Time Hypothesis is false. 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 value distances to be unitary (Wilder & Vorobeychik, 2018).

### 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).

While the reduction described in the proof of Theorem 1 does not work in this variant, we next show that it can be adapted to prove that ECS problem is inapproximable even in this setting.

**Theorem 4.** For any $\rho > 0$, there is no algorithm that, on input hard to manipulate instances of the ECS problem with bribed seeds and at least three candidates, always returns a $\rho$-approximate, 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, $\Delta_{MoV}(S^*,I^*,H) \geq (h + 1)\rho'$. Otherwise the only nodes that eventually change opinion are the seeds, that are at most $h + 1$. Thus, any $\rho$-approximation algorithm must be able to distinguish between these two cases and, therefore, it solves the Set-Cover problem in polynomial time. 

Instead, it is easy to check that Lemma 1 is unaffected by the fact that seeds are bribed, and thus a constant approximation is still possible when the number of hard-to-manipulate voters is $O(B)$ and there is at least a constant fraction of nodes that is uncertain.
3.3.2 Other Objective Functions

In addition to the maximization of the increase in the margin of victory, alternative objective functions, previously studied by Wilder and Vorobeychik (2018), 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 still holds. 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, Lemma 1 implies that a constant approximation can be computed in polynomial 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 by Wilder and Vorobeychik (2018). Hence, the problem cannot be approximated, within a factor $\rho > 0$, even for only two candidates, unless $P = NP$.

An apparently weaker goal would be to compute 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 compared to the setting described above). Unfortunately, this objective function does not make the problem easier to solve. 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 (Wilder & Vorobeychik, 2018).

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 (Kempe et al., 2015). This represents the most used diffusion model as an alternative to the independent cascade. In the linear threshold model, for each node $v$ of the network, there is a threshold $\theta_v$ 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, the results described in Theorem 1 and Corollary 1 still hold.

**Theorem 5.** Consider the variant of the ECS problem with at least three candidates in which information diffuses according to the threshold dynamics, and suppose that edge weights do not depend on the size of the graph. Then, for any $\rho > 0$, there is no algorithm that, on input instances of the ECS problem with $O(B)$ hard-to-manipulate voters, always returns a $\rho$-approximate solution, unless $P = NP$.

**Proof.** We consider a reduction from VERTEX-COVER, that 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 $Z$. By setting $n = gr - h$, we let components $G_2$ and $G_3$ have $n + h + 1$ and
Election Manipulation on Social Networks

$n + h + 3$ nodes, respectively. In particular, the nodes in each component $G_2$ and $G_3$ are arranged as directed rings (so that a message sent by a node in one component will activate all nodes in that component regardless of their threshold). Finally, we use the same initial ranks and same values as in the proof of Theorem 1 for nodes in $G_1$ and $G_2$. As for nodes in $G_3$, we have that $n + h + 2$ of them prefer $c_0$, and the remaining prefer $c_1$. Hence, in this configuration, $c_0$ and $c_1$ both take $n + g + 2$ votes. Finally, we set $B = rh + 1$.

If a vertex cover exists, then we seed the $h$ vertices forming this vertex cover in each of the $r$ copies of $Z$ with a positive message about $c_1$ and a single node in $G_2$ with a positive message about $c_2$. Hence, after the diffusion of messages, the number of voters for $c_1$ for $gr + 1 = n + h + 1$ (namely, all voters in $G_1$ and one voter in $G_3$), the number of voters for $c_2$ is $n + h + 1$ (namely, all voters in $G_2$), and the number of voters for $c_0$ is unchanged. Hence, the expected margin of victory of $c_0$ increases by 1.

Suppose, instead, that no vertex cover of size at most $h$ exists. As observed in the proof of Theorem 1, since we cannot increase the voters for $c_0$, the only way to have that the expected margin of victory of $c_0$ is larger than 0 would be to decrease the expected number of votes taken by $c_1$. This can only be done by injecting a positive message for $c_2$ (or, equivalently, a negative for $c_1$) in $G_2$. However, this makes all voters in $G_2$ vote for $c_2$.

Hence, it follows that we need to use the remaining $rh$ seeds in order to make all vertices from $G_1$ vote for $c_1$.

For each assignment of $rh$ seeds to nodes in $G_1$, there must be at least $r/(h+1)$ copies of $Z$ with at most $h$ seeds: indeed, if there is a fraction $x < 1/(h+1)$ of copies of $Z$ with at most $h$ seeds, then the number of allocated seeds should be at least $(1-x)r(h+1) > rh$. Since a vertex cover does not exist, there is a probability $p > 0$ that the nodes for which not all edges are covered by seeds do not change their mind. Hence, there are in expectation at least $\frac{rp}{h+1}$ nodes in $G_1$ that keep voting for $c_1$. Hence, the expected margin of victory of $c_0$ is in this case $(n + h + 2) - \left(n + h + 1 + \frac{rp}{h+1}\right) = 1 - \frac{r(1-p)}{h+1}$. By taking $r = \frac{(1-\epsilon)(h+1)}{1-p}$ (that is constant, according to our assumption on edge weights), we conclude that the margin of victory in this case is at most $\epsilon$. The theorem then follows.

For the case that the number of hard-to-manipulate voters is $O(B)$ and there is at least a constant fraction of voters that is uncertain, the greedy algorithm proposed in Lemma 1 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 (Kempe et al., 2015), and this is sufficient to make the proof of Lemma 1 and Corollary 1 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 by Khanna and Lucier (2014) for proving hardness of the influence maximization problem, from VERTEX-COVER in cubic graphs, i.e., the problem of finding a vertex cover in a graph in which every vertex has degree three.\(^{13}\)

Surprisingly, however, we have that, whenever the number of hard to manipulate voters is \(O(B)\) and there is at least a constant fraction of voters that is uncertain, a polynomial 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 polynomial-time algorithm for influence maximization returning a \(\left(1 - \frac{1}{\sqrt{e}} - \varepsilon\right)\)-approximation of the optimal seed set (Nguyen & Zheng, 2013). Then, the arguments of the proof of Lemma 1 and Corollary 1 immediately prove that this algorithm still provides an \(\Omega(1/\delta)\)-approximation for the extension of the election control problem to voters with different costs.

4. Edge Removal Complexity

We now focus on the ELECTION-CONTROL-BY-EDGE-REMOVAL (ECER) and INFLUENCE-MINIMIZATION-BY-EDGE-REMOVAL (IMER) problems. All the results provided in this section hold even with unitary value 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 \(P = NP\), as showed by Shieh et al. (2012).

**Definition 8 (Maximum-Subset-Intersection (MSI)).** Given a set \(N = \{z_1, \ldots, z_n\}\) of elements, a collection \(X = \{x_1, \ldots, x_g\}\) of sets with \(x_i \subset N\), and a positive integer \(h\), the goal is to find exactly \(h\) subsets \(x_{j_1}, \ldots, x_{j_h}\) whose intersection size \(\left|\bigcap_{i=1}^{h} x_{j_i}\right|\) is maximum.

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

**Proof.** We reduce from MSI, showing that a constant-factor approximation algorithm for IMER implies that a constant-factor polynomial-time approximation for MSI exists, thus having a contradiction unless \(P = NP\). 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 \(j \in \{1, \ldots, n^2 g^2\}\). 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 \(j \in \{1, \ldots, n^2 g^2\}\), i.e., all the nodes of all the elements not in the set \(x_i\). Figure 7 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 choosing \(g - h\) sets \(x_i \in X\) and removing the edges from \(v_{x_i,1}\) to \(v_{x_i,2}\), such that as few nodes \(v_{z_i,j}\) as possible are influenced. The optimal value is

\(^{13}\) Note that, in that reduction, the authors assume that there is a set of non-allowed seeds, 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.
obtained by choosing \( X^* \subset X \) of cardinality \( h \) that is solution of MSI and then removing the edges from \( v_{x_1,1} \) to \( v_{x_1,2} \) for all \( x \in X \setminus X^* \). Call this set of edges \( E^* \). If we remove the edges in \( E^* \), all the nodes \( v_{z_i,j}, z_i \in \cap_{x_1 \in X} x_2 \) are not influenced, since there are no edges from \( v_{x_1,2}, x \in X^* \) to \( v_{z_i,j} \), as they all exist in the complement of the bipartite graph.

The relationship between the IMER\’s optimal solution and the one of MSI is \( \Delta I - (E^*) = g - h + OPT n^2 g^2 \), where \( OPT \) is the optimal solution to MSI. Assume for sake of contradiction there exists an \( \rho \)-approximation algorithm \( A \) 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 \( \rho' \) such that \( APX \geq \rho' OPT \) and an algorithm \( A' \) for MSI with a \( \rho' \)-approximation factor.

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

**Corollary 2.** For any constant \( \rho > 0 \), there is no polynomial time algorithm returning a \( \rho \)-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 as in the proof of Theorem 6, two candidates, all nodes with values \((1, 0)\) and seeds with messages \((-1, 0)\). It is easy to see that \( \Delta_{M_0V} = 2\Delta I^- \). Since approximating \( \Delta I^- \) is hard, it follows that approximating \( \Delta_{M_0V} \) 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 allowing the diffusion of different messages makes the problem hard. We introduce the Independent-Set problem, that is known to be inapproximable within any constant factor by Zuckerman (2007)\(^\text{14}\). We reduce from it to prove the hardness of the ECER when there are three candidates and messages are single-news-article.

**Definition 9 (Independent-Set).** Given a graph \(G = (X, N)\), with \(|X| = g\) vertices and \(|N| = n\) edges, find the largest set of vertices \(X^*\) such that there is no edge connecting two vertices in \(X^*\).

**Theorem 7.** For any constant \(\rho > 0\), there is no polynomial time algorithm returning a \(\rho\)-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 8 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.

\(^\text{14}\) Actually, Independent-Set is much harder to approximate than this. However, constant factor inapproximability is sufficient for our results.
Figure 8: 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 \( c_1 \) looses 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 \( \rho \)-approximation algorithm \( A \) 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 \( A \). Since \( \Delta_{\text{MoV}}(E') > 0 \), \( A \) 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 \( \rho \)-approximation algorithm for ECER, there exists a \( \rho \)-approximation algorithm for INDEPENDENT-SET, leading to a contradiction.

We now focus on instances in which messages can be arbitrary and with 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 \( \rho \)-approximation to the ECER even with two candidates and unlimited budget, unless \( P = NP \).

**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 \((1, 0)\) and seeded with messages \( q_0 = 1 \) and \( q_1 = -1 \), a node \( v_2 \) with preferences \((1, 0)\) 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 \((1, 0)\) 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 9 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 \( \Delta_{\text{MoV}} \) is larger than 0 if and only if there is a set cover of size \( h \).

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 \( x_i \in X \setminus X^* \), where \( X^* \) is the optimal solution to the instance of \( \text{Set-Cover} \). 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 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 vote 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 is 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 for sake of 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 Election-Control-by-Edge-Addition (ECEA) and Influence-Maximization-by-Edge-Addition (IMEA) problems. All the results provided in this section hold even with unitary value 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 \( P = \text{NP} \), while influence maximization by seeding can be. In our proof, we reduce from the maximization version of Set-Cover, called Max-Cover.

**Definition 10 (Max-Cover).** Given a finite set \( N = \{z_1, \ldots, z_n\} \) of elements, a collection \( X = \{x_1, \ldots, x_g\} \) of sets with \( x_i \subset N \), and \( h \in N^+ \), the objective is to select \( X^* \subset X \), with \( |X^*| \leq h \), that maximizes \( |\bigcup_{x_i \in X^*} x_i| \).

Feige (1998) proves that, given an instance of Max-Cover, deciding whether all the elements can be covered or at most a \((1 - \frac{1}{e} + \epsilon)\) fraction of them can be covered is \( \text{NP}-\text{hard} \) for any \( \epsilon > 0 \).

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

**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 \( \{1, \ldots, n^8\} \), 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 \( v_{i,z_t} \) with probability 1 and an edge from \( v_{i,z_t} \) to \( v_{i+1} \) with probability \( 1 - \frac{1}{n^8} \). We add a node \( v_{n^8+1} \) and an edge with probability 1 towards \( n^10 \) further nodes.

Call the subgraph composed by these \( n^10 \) nodes \( G' \). The resulting graph is depicted in Figure 10. The only seed is \( v_1 \), and the only edges that can be added are the edges between \( v_i \) and \( v_{i,x_j} \) with probability one. The budget is \( h \cdot n^8 \). Suppose Max-Cover is satisfiable, i.e., there exists a set \( X^* \) that covers all the elements. Consider the set of edges \( E^* \) including, for each \( i \in \{1, \ldots, n^8\} \), the edges from \( v_i \) to all \( v_{i,x_j} \) for all \( x_j \in X^* \). Adding the edges in \( E^* \), 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.,

\[
\Delta I^+ > \left[ 1 - \left( \frac{1}{n^8} \right) \right] n^8 n^{10} > \left( \frac{1}{e} - \epsilon \right) n^{10}
\]

for all \( \epsilon > 0 \) and \( n \) large enough. Suppose each cover of size at most \( h \) covers at most \( \frac{3}{4} \) of the elements. 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 \( \frac{1}{4} \) vertices \( v_{i,z} \) without incoming edges. Thus the probability of activating \( G' \) is smaller than

\[
\left[ 1 - \left( \frac{1}{n^8} \right)^n \right]^{\frac{n^8}{n+1}} \leq \frac{1}{e^{n^8} e^{n^8}} = \frac{1}{e^{n^8}} \leq e^{-n}.
\]

and \( \Delta I^+(E') \leq n^8(n + g + 1) + 1 + e^{-n}\). Clearly \( \frac{\Delta I^+(E')}{\Delta I^+(E^*)} < \rho \), for each \( \rho > 0 \) and \( n \) sufficiently large. Hence, a \( \rho \)-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.

Figure 10: Structure of the election control problem used in the proof of Theorem 9.

Now we can state the following result, whose proof follows directly from the proof of the above theorem.

**Corollary 3.** For any constant \( \rho > 0 \), there is no polynomial time algorithm returning a \( \rho \)-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 no polynomial time algorithm returning a $\rho$-approximation to the ECEA with single-news-article messages even when there are three candidates and the budget is unlimited, unless $\mathbb{P} = \mathbb{NP}$.

**Proof.** Given an instance of SET-COVER, we build an instance of election control as follows. We add a node $v_1$ with preferences $(0, 1, 2)$ and seed it with $q_2 = 1$. We add a line $L_1$ of $ng - h - 1$ nodes with preferences $(2, 0, 1)$. We add a node $v_{x_i}$ for each set $x_i \in X$ with preferences $(0, 2, 1)$. For each element $z_i \in N$, we add a line $L_{z_i}$ of $g$ nodes with ranking $(0, 2, 1)$ and an edge from each $x_j \supseteq z_i$ to the first node of $L_{z_i}$. Moreover, we add $n^2g^2$ isolated nodes with preferences $(2, 1, 0)$ and $n^2g^2$ isolated nodes with preferences $(1, 2, 0)$.

An example of the network produced with the above mapping is depicted in Figure 11. The only edges that can be added are the edge 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 11: Structure of the election control problem used in the proof of Theorem 10.](image)

**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 non-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 no polynomial time algorithm returning a $\rho$-approximation to the ECEA even with two candidates and unlimited budget, unless $\mathbb{P} = \mathbb{NP}$.
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 $x_i \in X$ 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 12: Structure of the election control problem used in the proof of Theorem 11.](image)

Notice that, if no edges are added, no voter changes her votes and 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, since the only seeds are $v_1$ and $v_2$, 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 $L_{z_i}$ 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. \qed
6. Reoptimization Complexity

In this section, we show a form of robustness of our hardness results. Specifically, we consider a reoptimization setting, in which the manipulator already knows a solution to the problem, and it is asked to compute a solution to a variant of this original problem. In particular, the local modification that we consider in our work is a very weak one (thus making our hardness results even more robust): we change the transmission probability of a single edge. Note that this modification in the input is very common: it may arise because previous probabilities were not precise, and more precise estimate are now available, or because the relationship among voters is changed.

Formally, we first consider the following reoptimization problem.

**Definition 11.** An election control through social influence by seeding reoptimization problem $S\text{ReOpt}(I, S^*, e, o)$ 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 no polynomial time algorithm returning a $\rho$-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 $v_{z_i} \in x_i$. Finally, we add a node $v^*$ with an edge from $v^*$ to all nodes $v_{z_i}, z \in Z$, and an edge from $v^*$ to $v_{x_i,2}$ (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 $v_{z_i}$, any $h$ nodes $v_{x_i,1}$ and a node in $G_2$. Consider the problem $S\text{ReOpt}(I, S^*, (v^*, v_{x_i}), 0)$, its optimal solution is the optimal solution of the optimization problem over $I$. If $S\text{ReOpt}(I, S^*, (v^*, v_{x_i}), 0)$ can be approximated in polynomial time, then Set-Cover can be solved in polynomial time. □

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

**Definition 12.** An election control through social influence by edge removal or addition reoptimization problem $E\text{ReOpt}(I, E^*, e, o)$ 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 $I$. 

1085
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 
\[ \max_v \{ \max_{c_i} \pi_v(i) - \min_{c_i} \pi_v(i) \} = O(\text{poly(size}(I))) \], reoptimization is as hard as optimization.

**Proof.** Consider an instance \( I \) of election control with \( G = (V, E, p) \). By assumption \( d = \max_v [\max_{c_i} \pi_v(i) - \min_{c_i} \pi_v(i)] = O(\text{poly(size}(I))) \), i.e., \( d \) is polynomially upper bounded in the instance size. We build a graph \( G_1 \) with \( d + 1 \) nodes \( \{v_i\}, i \in \{0, \ldots, d\} \) 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 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 \( EReOpt(I, \emptyset, (v_1^*, v_2^*), 0) \): its optimal solution is the optimal solution of the optimization problem over \( I \).

We remark that 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 study the problem of manipulating the outcome 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 use 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 polynomial time. Our main result shows that the election manipulation problem is not affordable in the worst-case, even when one accepts to get an approximation of the optimal margin of victory that is a polynomial function in the size of the instance, except for the case of seeding when the number of hard-to-manipulate voters is a sublinear function in the budget available to the manipulator. In real-world elections, it is reasonable that some voters will never change opinion, e.g., the supporters of the other candidates. Our result shows that if these voters are not too many with respect to the budget available to the manipulator, then a constant approximation of the margin of victory can be achieved. From a prescriptive point of view, our result provides the manipulator with a lower bound on the budget to guarantee the election’s manipulability. On the opposite side, to guarantee the non-manipulability of the election, the central authority could use our result to pose an upper bound to the electoral campaigns’ costs motivated by, e.g., fairness arguments.

Furthermore, we show that the most known heuristic algorithms for social influence maximization (e.g., greedy, degree centrality, PageRank, VoteRank) do not provide any
approximation factor even with basic graphs. This result is crucial, remarking the need for completely novel heuristics for the election manipulation problem. 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 for a reoptimization variant.

While we provided a polynomial 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. Furthermore, it would be interesting to analyze other generalizations of our model, such as different models for information diffusion and time-evolving networks — see, e.g., the work of Auletta et al. (2019a) and references therein. Finally, theoretical and experimental analysis of the problems on realistic networks would be of extreme interest.

Acknowledgments

Castiglioni et al. (2020b) provide an extended abstract of this work. Moreover, Castiglioni et al. (2020a) include some results on edge addition/removal. This work has been partially supported by the Italian MIUR PRIN 2017 Project ALGADIMAR “Algorithms, Games, and Digital Market”.

References


Election Manipulation on Social Networks


Castiglioni, Ferraioli, Gatti, & Landriani


