Voter Participation Control in Online Polls

Extended Abstract

Koustav De
IIT Kharagpur
Kharagpur, India
koustavde7@kgpian.iitkgp.ac.in

Palash Dey
IIT Kharagpur
Kharagpur, India
palash.dey@cse.iitkgp.ac.in

Swagato Sanyal*
University of Sheffield
Sheffield, UK
swagato.sanyal@sheffield.ac.uk

ABSTRACT

News outlets, surveyors, and other organizations often conduct polls on social networks to gain insights into public opinion. Such a poll is typically started by someone on a social network who sends it to her friends. If a person participates in the poll, the poll information gets published on her wall, which in turn enables her friends to participate, and the process continues. Eventually, a subset of the population participates in the poll, and the pollster learns the outcome of that poll. We initiate the study of a new but natural type of election control in such online elections.

We study how difficult/easy it is to sway the outcome of such polls in one's favor/against (aka constructive vs destructive) by any malicious influencer who nudges/bribes people for seemingly harmless actions like non-participation. These questions are important from the standpoint of studying the power of resistance of online voting against malicious behavior.

KEYWORDS

Voting theory, social choice, plurality voting rule, graph algorithms, NP-hardness, bounded treewidth graphs, parameterized algorithms.

ACM Reference Format:
Koustav De, Palash Dey, and Swagato Sanyal. 2025. Voter Participation Control in Online Polls: Extended Abstract. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 3 pages.

1 INTRODUCTION

Voting is arguably the most widely used tool when a set of people needs to decide on one alternative/candidate [19, 24]. The study of various election malpractices and their complexity are one of the core research focuses in computational social choice [1, 4].

This work focuses on online voting where there is a social network on the set of voters. A voter initiates an election (online poll or survey); *e.g.*, on a Facebook network, a voter can post a poll on her wall that, when her friends see it, they participate in it. When their friends participate in that poll, their friends will see it and can participate in it. If everyone who sees the poll participates in it, then, if the social network is connected, then everyone in the network participates in it. However, this is rarely the case because some people may not participate in the poll even after seeing it. Hence, only a connected subset of people, including the poll initiator, participates in that poll. We study an important type of attack on such elections: persuading some voters not to participate in the election, thereby controlling the set of voters that eventually participate in the election. We primarily focus on the objective of helping some candidate win/lose the election under a complexity-theoretic lens.

2 RELATED WORK

Bartholdi et al. [1] initiated the study of electoral control. The chair may exercise control over the candidate set by removing up to k candidates from the election or by inserting new candidates from a list of spoiler candidates. Hemaspaandra et al. [17] later defined a variant of the problem where the number of spoiler candidates that might be added by the election controller has a bound k. Many voting rules, for example, Fallback and Bucklin [9], Copeland$^\alpha$ [12], Normalized Range Voting [21], SP-AV voting [10] and Schulze Voting [22] are resistant to all types of constructive control. Bodlaender [2] showed that intractable computational problems on graphs usually become tractable if the treewidth is bounded by a constant. Slinko and White examined the class of social choice functions that can be safely manipulable [26, 27]. Hazon and Elkind [16] and Ianovski et al. [18] have looked into the complexity of safely manipulating popular voting rules. Faliszewski et al. [11] have shown that in a bribery problem, a briber, who can be an election controller, can change the minimum number of preferences to make way for a preferred candidate to win the election. Bredereck and Elkind [5] analysed the computational complexity of bribery and control by adding/deleting links between users on an online social network and altering the order in which voters update their opinions. Goles and Olivos [14] showed that a sequence of at most $O(n^2)$ synchronous updates, where n is the number of voters, always converges to a stable state. Frischknecht et al. [13] strengthened the tightness of the stated result. Wilder and Vorobeychik [29] showed hardness, inapproximability, and algorithmic results for constructive and destructive control. Opinion dynamics and social choice have been extensively studied by [6, 15, 23, 25, 28].

3 PRELIMINARIES AND DEFINITIONS

We refer to the full version of our paper for all the details [8].

Problem Definition 1 (Constructive and Destructive Control over Network). *We are given a set $C = \{c_1, \ldots, c_m\}$ of m candidates, a set $\mathcal{V} = \{v_1, \ldots, v_n\}$ of n voters, a voting function $\tau : \mathcal{V} \to C$, an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ whose vertices are the voters, a target candidate $c \in C$ of the controller, a voter $x \in \mathcal{V}$ who conducts the election, a cost function $\pi : \mathcal{V} \longrightarrow \mathbb{R}_{\geq 0}$, and a budget $\mathcal{B}$ of the controller. We extend the definition of π to subsets of $\mathcal{V}$, and define the cost $\pi(\mathcal{K})$ of a subset $\mathcal{K} \subseteq \mathcal{V}$ as $\sum_{v \in \mathcal{K}} \pi(v)$. We say that*

*When this work was completed, Swagato Sanyal was in IIT Kharagpur.

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

a subset $\mathcal{W} \subseteq \mathcal{V} \setminus \{x\}$ is "budget feasible" if $\sum_{v \in \mathcal{W}} \pi(v) \leq \mathcal{B}$. For a budget feasible set $\mathcal{W}$, let $\mathcal{H}_{\mathcal{W}}$ be the set of nodes reachable from x in $\mathcal{G} \setminus \mathcal{W}$.

We consider the plurality voting rule in this paper. A candidate $c \in C$ is declared a winner of the election if $c \in \arg\max_{c' \in C} |\{v \in \mathcal{V} \mid \tau(v) = c'\}|$. If there is only one winner c of an election, then c is said to win the election uniquely. Constructive Control over Network problem asks whether there exists a budget feasible set $\mathcal{W}$ such that c wins uniquely in the restricted election $(C, \mathcal{V}, \tau)$ where only votes of $\mathcal{H}_{\mathcal{W}}$ are counted. In contrast, Destructive Control over Network problem asks whether there exists a budget feasible set $\mathcal{W}$ such that a candidate other than c unambiguously wins in the restricted election $(C, \mathcal{V}, \tau)$ where only votes of $\mathcal{H}_{\mathcal{W}}$ are counted. Both Constructive Control over Network and Destructive Control over Network take tuple $(C, \mathcal{V}, \tau, \mathcal{G}, c, x, \pi, \mathcal{B})$ as generic input. We also consider the special setting where the budget is infinite. In this setting, the cost function π is irrelevant, and any subset $\mathcal{W} \subseteq \mathcal{V} \setminus \{x\}$ trivially satisfies the budget constraint. We refer to the corresponding versions as Budgetless Constructive Control over Network and Budgetless Destructive Control over Network. An instance of the budgetless versions is a tuple $(C, \mathcal{V}, \tau, \mathcal{G}, c, x)$, where the members of the tuple are as per Problem Definition 1. Observe that an efficient algorithm $\mathcal{A}$ for Constructive Control over Network can be used to design an efficient algorithm for Destructive Control over Network as follows: examine each candidate c' other than c in turn, and decide whether c' can be made a unique winner within the given budget by invoking $\mathcal{A}$. Regular Exact 3-cover, defined below, is known to be NP-complete [20].

*Definition 3.1 (*Regular Exact 3-cover*).* For a positive integer ℓ, let $\mathcal{U} \coloneqq \{1, \ldots, 3\ell\}$. We are given m subsets $S_1, \ldots, S_m$ of $\mathcal{U}$, each of cardinality 3 such that $\cup_{i \in [m]} S_i = \mathcal{U}$. Furthermore, each element in $\mathcal{U}$ belongs to exactly two sets in the collection $\{S_1, \ldots, S_m\}$. Decide whether there exists a subset $A \subseteq [m]$ of size ℓ such that $\cup_{i \in A} S_i = \mathcal{U}$. Note that if such a set A exists, then the sets $\{S_i \mid i \in A\}$ are pairwise disjoint. We denote an instance of Regular Exact 3-cover as $(\ell, S_1, \ldots, S_m)$.

3.1 Tree decomposition and treewidth

We refer to the full version of our paper for all the details [8]. Also see [7] for more details.

4 OUR CONTRIBUTION

We study the computational complexity of Constructive Control over Network and Destructive Control over Network.

Our first result shows that Destructive Control over Network admits a polynomial time algorithm for the special case where the treewidth (see Section 3.1 for related definitions) of the graph $\mathcal{G}$ is a constant.

Theorem 4.1. *There exists an algorithm that, given an input $(C, \mathcal{V}, \tau, \mathcal{G}, c, x, \pi, \mathcal{B})$ of* Destructive Control over Network *and a tree decomposition T of the graph $\mathcal{G}$ of width w, solves* Destructive Control over Network *in time $w^{O(w)} \cdot n^{O(w)} \cdot \text{poly}(n, m)$. In particular,* Destructive Control over Network *admits a polynomial time algorithm when $\mathcal{G}$ is a tree.*

How about Constructive Control over Network? As discussed in Section 3, Constructive Control over Network is computationally at least as hard as Destructive Control over Network (up to a factor of m). Our next result obtains a polynomial time algorithm for Constructive Control over Network with an assumption of the treewidth of $\mathcal{G}$ being a constant as in Theorem 4.1, and an additional assumption that the number of candidates m is a constant.

Theorem 4.2. *There exists an algorithm that, given an input $(C, \mathcal{V}, \tau, \mathcal{G}, c, x, \pi, \mathcal{B})$ of* Constructive Control over Network *and a tree decomposition T of the graph $\mathcal{G}$ of width w, solves* Constructive Control over Network *in time $w^{O(w)} \cdot n^{O(mw)} \cdot \text{poly}(n, m)$. In particular,* Constructive Control over Network *admits a polynomial time algorithm when $\mathcal{G}$ is a tree, and the number of candidates m is a constant.*

Theorem 4.1 and Theorem 4.2 assume that a tree decomposition of low treewidth is given as a part of the input. However, this assumption is not restrictive; it is known that given a graph $\mathcal{G}$ with n vertices it is possible to construct a tree decomposition of $\mathcal{G}$ of width $\mathbf{tw}(\mathcal{G})$ in time $O(f(\mathbf{tw}(\mathcal{G})) \cdot n)$, where $f(\cdot)$ is a quasi-polynomially growing function [3, 7].

Are the assumptions in Theorem 4.1 and Theorem 4.2 necessary? We answer this question in the affirmative by two hardness results. The first one shows that Constructive Control over Network is NP-complete even when the graph $\mathcal{G}$ is a tree (i.e. has treewidth 1) and the setting is budgetless.

Theorem 4.3. Budgetless Constructive Control over Network *is NP-complete even if the input graph $\mathcal{G}$ is a tree.*

Our next result shows that both the problems are NP-complete even in the special case where there are two candidates and the setting is budgetless.

Theorem 4.4. Budgetless Constructive Control over Network *and* Budgetless Destructive Control over Network *are NP-complete even for the special case where there are two candidates (i.e. $m = 2$).*

Observe that for $m = 2$ the two problems are equivalent in the following sense. Let $C = \{0, 1\}$. For each $b \in C$, $(C, \mathcal{V}, \tau, \mathcal{G}, b, x, \pi, \mathcal{B})$ is a YES instance of Constructive Control over Network *if and only if* $(C, \mathcal{V}, \tau, \mathcal{G}, 1 - b, x, \pi, \mathcal{B})$ is a YES instance of Destructive Control over Network. Thus, in order to prove Theorem 4.4 it suffices to prove the NP-hardness of Constructive Control over Network for $m = 2$ in the budgetless setting.

Our hardness results presented in Theorem 4.3 and Theorem 4.4 thus complement our algorithmic results presented in Theorem 4.1 and Theorem 4.2, and establish the unavoidability of the assumptions made in them. This paper presents a comprehensive study of the computational complexity of Constructive Control over Network and Destructive Control over Network.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their valuable suggestions. Palash Dey thanks the Science, Education, and Research Board (SERB), Government of India, for supporting this work through Core Research Grant under file no. CRG/2022/003294.

REFERENCES

[1] John J. Bartholdi, Craig A. Tovey, and Michael A. Trick. 1992. How hard is it to control an election? *Mathematical and Computer Modelling* 16, 8 (1992), 27–40. https://doi.org/10.1016/0895-7177(92)90085-Y

[2] Hans L. Bodlaender. 1988. Dynamic Programming on Graphs with Bounded Treewidth. In *Proceedings of the 15th International Colloquium on Automata, Languages and Programming (ICALP '88)*. Springer-Verlag, 105–118.

[3] Hans L. Bodlaender. 1996. A Linear-Time Algorithm for Finding Tree-Decompositions of Small Treewidth. *SIAM J. Comput.* 25, 6 (1996), 1305–1317. https://doi.org/10.1137/S0097539793251219

[4] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia (Eds.). 2016. *Handbook of Computational Social Choice.* Cambridge University Press. https://doi.org/10.1017/CBO9781107446984

[5] Robert Bredereck and Edith Elkind. 2017. Manipulating Opinion Diffusion in Social Networks. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 894–900. https://doi.org/10.24963/ijcai.2017/124

[6] Vincent Conitzer. 2013. The maximum likelihood approach to voting on social networks. In *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. 1482–1487. https://doi.org/10.1109/Allerton.2013.6736702

[7] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. 2015. *Parameterized Algorithms*. Springer. https://doi.org/10.1007/978-3-319-21275-3

[8] Koustav De, Palash Dey, and Swagato Sanyal. 2025. Voter Participation Control in Online Polls. arXiv:2410.12256 [cs.MA] https://arxiv.org/abs/2410.12256

[9] Gábor Erdélyi and Jörg Rothe. 2010. Control Complexity in Fallback Voting. In *Theory of Computing 2010, CATS 2010 (CRPIT, Vol. 109)*. Australian Computer Society, 39–48.

[10] Gábor Erdélyi, Markus Nowak, and Jörg Rothe. 2009. Sincere-Strategy Preference-Based Approval Voting Fully Resists Constructive Control and Broadly Resists Destructive Control. *Mathematical Logic Quarterly* 55, 4 (2009), 425–443. https://doi.org/10.1002/malq.200810020

[11] Piotr Faliszewski, Edith Hemaspaandra, and Lane A. Hemaspaandra. 2006. The Complexity of Bribery in Elections. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006*. AAAI Press, 641–646.

[12] Piotr Faliszewski, Edith Hemaspaandra, Lane A. Hemaspaandra, and Jörg Rothe. 2009. Llull and Copeland voting computationally resist bribery and constructive control. *J. Artif. Int. Res.* 35, 1 (jun 2009), 275–341.

[13] Silvio Frischknecht, Barbara Keller, and Roger Wattenhofer. 2013. Convergence in (Social) Influence Networks. In *Distributed Computing*, Vol. 8205. Springer Berlin Heidelberg, 433–446.

[14] Eric Goles and J. Olivos. 1980. Periodic behaviour of generalized threshold functions. *Discret. Math.* 30, 2 (1980), 187–189. https://doi.org/10.1016/0012-365X(80)90121-1

[15] Umberto Grandi. 2017. Social Choice and Social Networks. In *Trends in Computational Social Choice*, Ulle Endriss (Ed.). AI Access, Chapter 9, 169–184.

[16] Noam Hazon and Edith Elkind. 2010. Complexity of Safe Strategic Voting. In *Algorithmic Game Theory*. Springer Berlin Heidelberg, 210–221.

[17] Edith Hemaspaandra, Lane A. Hemaspaandra, and Jörg Rothe. 2009. Hybrid Elections Broaden Complexity-Theoretic Resistance to Control. *Mathematical Logic Quarterly* 55, 4 (2009), 397–424. https://doi.org/10.1002/malq.200810019

[18] Egor Ianovski, Lan Yu, Edith Elkind, and Mark C. Wilson. 2011. The Complexity of Safe Manipulation under Scoring Rules. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, July 16-22, 2011*. IJCAI/AAAI, 246–251. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-052

[19] Benjamin G. Jackson, Patrick S. Schnable, and Srinivas Aluru. 2008. Consensus Genetic Maps as Median Orders from Inconsistent Sources. *IEEE ACM Trans. Comput. Biol. Bioinform.* 5, 2 (2008), 161–171. https://doi.org/10.1109/TCBB.2007.70221

[20] Pak Ching Li and Michel Toulouse. 2006. Variations of the maximum leaf spanning tree problem for bipartite graphs. *Inf. Process. Lett.* 97, 4 (2006), 129–132. https://doi.org/10.1016/J.IPL.2005.10.011

[21] Curtis Menton. 2013. Normalized Range Voting Broadly Resists Control. *Theor. Comp. Sys.* 53, 4 (nov 2013), 507–531. https://doi.org/10.1007/s00224-012-9441-0

[22] Curtis Glen Menton and Preetjot Singh. 2013. Control Complexity of Schulze Voting. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, August 3-9, 2013*. IJCAI/AAAI, 286–292.

[23] Elchanan Mossel, Joe Neeman, and Omer Tamuz. 2014. Majority dynamics and aggregation of information in social networks. *Autonomous Agents and Multi-Agent Systems* 28, 3 (may 2014), 408–429. https://doi.org/10.1007/s10458-013-9230-4

[24] David M. Pennock, Eric Horvitz, and C. Lee Giles. 2000. Social Choice Theory and Recommender Systems: Analysis of the Axiomatic Foundations of Collaborative Filtering. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on on Innovative Applications of Artificial Intelligence, July 30 - August 3, 2000*. AAAI Press / The MIT Press, 729–734.

[25] Ariel D. Procaccia, Nisarg Shah, and Eric Sodomka. 2015. Ranked voting on social networks. In *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15)*. AAAI Press, 2040–2046.

[26] Arkadii Slinko and Shaun White. 2008. Nondictatorial social choice rules are safely manipulable. *COMSOC* 8 (2008), 403–413.

[27] Arkadii Slinko and Shaun White. 2014. Is it ever safe to vote strategically? *Soc. Choice Welf.* 43, 2 (2014), 403–427. https://doi.org/10.1007/S00355-013-0785-4

[28] Alan Tsang and Kate Larson. 2016. The Echo Chamber: Strategic Voting and Homophily in Social Networks. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems (AAMAS '16)*. International Foundation for Autonomous Agents and Multiagent Systems, 368–375.

[29] Bryan Wilder and Yevgeniy Vorobeychik. 2018. Controlling Elections through Social Influence. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS '18)*. International Foundation for Autonomous Agents and Multiagent Systems, 265–273.