Fighting Opinion Control in Social Networks via Link Recommendation*

(Supplementary Materials)

Victor Amelkin
Warren Center for Network and Data Sciences
University of Pennsylvania
vctr@seas.upenn.edu

Ambuj K. Singh
Department of Computer Science
University of California, Santa Barbara
ambuj@cs.ucsb.edu

PROOF OF NP-HARDNESS

THEOREM 3. *Problem* DIVER$(W, k, x, \widetilde{x})$ *is NP-hard for undirected networks.*

PROOF. In the proof, we will show that

$$\mathrm{DIVER}(W, k, x, \widetilde{x}) = \arg\min_{\widetilde{W}} |\langle \widetilde{\pi}(\widetilde{W}), \widetilde{x}\rangle - \langle \pi, x\rangle|$$

applied to a certain simple undirected network can be used as a solver for the classic NP-complete subset sum problem. For readability, we will abuse notation and assume that DIVER$(W, k, x, \widetilde{x})$ is the minimum itself, rather than the corresponding arg min.

1) Subset sum problems: The subset sum problem SSP$(\{z_i\}, s)$ is a classic NP-complete problem of deciding whether a given finite set $\{z_i\} \subset \mathbb{Z}^n$ of integers has a non-empty subset with a predefined sum $s \in \mathbb{Z}$. (SSP appears on Karp's list of NP-complete problems [5, p.95] under the name KNAPSACK). A related problem is the problem kSSP$_{01}(\{z_i\}, k, s)$ of deciding whether, among a finite number of bounded reals $z_i \in [0, 1]$, there is a non-empty subset of k elements summing up to a given value $s \in [0, 1]$. Reduction SSP $\propto$ kSSP$_{01}$ is as follows:

$$\begin{aligned}
&\mathrm{SSP}(\{z_i\}, s) = \bigvee\nolimits_{k=1}^{n} \mathrm{kSSP}_{01}(\{z_i''\}, k, s''),\\
&z_i' = z_i + L \in \mathbb{Z}_+,\ s' = s + kL \in \mathbb{Z}_+,\\
&L = |\min\{0, \min\{s, \min z_i\}\}|\\
&z_i'' = z_i'/M \in [0, 1],\ s_i'' = (s + kL)/M \in [0, 1],\\
&M = \max\{s', \max z_i'\}.
\end{aligned}$$

2) Undirected uniformly weighted networks and their eigenvector centrality: Let $W_{01} \in \{0, 1\}^{n\times n}$ be the binary adjacency matrix of an undirected network, $d = W_{01}\mathbb{1}$ be a vector of node degrees, and $D = \mathrm{diag}(d)$. Further, let $W = D^{-1}W_{01}$. We say that W is the adjacency matrix of an *undirected uniformly weighted network* (since, all the edges within the same neighborhood are weighted equally). Notice that W is row-stochastic, as $W\mathbb{1} = D^{-1}W_{01}\mathbb{1} = D^{-1}d = \mathbb{1}$.

Since $d^\intercal W = d^\intercal D^{-1}W_{01} = \mathbb{1}^\intercal W_{01} = d^\intercal$, vector $\pi = d/\|d\|_1 = d/(2|E|) = d/(2m)$ is the ℓ_1-normalized dominant left eigenvector–or, eigenvector centrality—of W. If the underlying unweighted network W_{01} is perturbed with k undirected edges $(i, j) \in S$, $|S| = k$, then the eigenvector centrality of the corresponding weighted network becomes

$$\begin{aligned}
\widetilde{\pi} &= \frac{1}{2(m+k)}\left(d + \sum\nolimits_{(i,j)\in S}(e_i + e_j)\right)\\
&= \frac{1}{m+k}\left(m\pi + \sum\nolimits_{(i,j)\in S}(e_i + e_j)/2\right), \qquad (1)
\end{aligned}$$

where e_i is the i'th column of the identity matrix.

3) DIVER *in undirected uniformly weighted networks:* If network W is undirected uniformly weighted and, thus, defined by its binary adjacency matrix W_{01}, then DIVER's objective function over such W can be rewritten as follows:

$$\begin{aligned}
f(\widetilde{W}_{01}) &= |\langle \widetilde{\pi}, \widetilde{x}\rangle - \langle \pi, x\rangle| = \text{(from (1))} =\\
&= \left|\frac{m}{m+k}\langle \pi, \widetilde{x}\rangle + \frac{1}{2(m+k)}\langle \sum_{(i,j)\in S}(e_i + e_j), \widetilde{x}\rangle - \langle \pi, x\rangle\right|\\
&= \frac{1}{m+k}\left|\langle \sum_{(i,j)\in S}(e_i + e_j), \widetilde{x}/2\rangle - \langle \pi, (m+k)x - m\widetilde{x}\rangle\right|\\
&= a(k)\left|\langle \sum_{(i,j)\in S}(e_i + e_j), \widetilde{x}/2\rangle - b(k, x, \widetilde{x})\right|,
\end{aligned}$$

where $b(k, x, \widetilde{x}) = \langle \pi, (m+k)x - m\widetilde{x}\rangle$. Since k, and, consequently, $a(k)$ are constant, minimization of $f(\widetilde{W}_{01})$ is equivalent to minimization of

$$f'(\widetilde{W}_{01}) = \left|\langle \sum_{(i,j)\in S}(e_i + e_j), \widetilde{x}/2\rangle - b(k, x, \widetilde{x})\right|. \qquad (2)$$

4) Reduction kSSP$_{01}$ $\propto$ DIVER*:* Suppose we are given an instance kSSP$_{01}(z, k, s)$, with $z \in [0, 1]^n$, $k \in \mathbb{N}$, and $s \in [0, 1]$. In what follows, we will show that the solution to kSSP$_{01}(z, k, s)$ is obtained by checking whether

$$\min_{\widetilde{W}^{KC}} \mathrm{DIVER}\left(W^{KC}, k, \frac{s\mathbb{1} + m(z \otimes \mathbb{1}_2)}{m+k}, z \otimes \mathbb{1}_2\right) = 0, \qquad (3)$$

where W^{KC} is the adjacency matrix of an undirected uniformly weighted $2n$-clique from which edges $C = \{(2i-1, 2i) \mid i = 1, \ldots, n\}$ have been removed (see Fig. 1), $\widetilde{W}^{KC}$ is W^{KC} perturbed with k edges $S = \{(2i-1, 2i)\} \subseteq C$, $\mathbb{1} = \mathbb{1}_{2n}$, and $\otimes$ is Kronecker product.

It is easy to show that the proposed input to DIVER is indeed legal (W^{KC} is row-stochastic matrix of a uniformly weighted undirected strongly connected aperiodic network; and $\widetilde{x} = z \otimes \mathbb{1}_2$ and $x = \frac{s\mathbb{1}+m(z\otimes\mathbb{1}_2)}{m+k}$ are legal vectors of altered and original user opinions, respectively.

This is supplementary material for work [1]. The work was supported by U.S. Army Research Laboratory and U. S. Army Research Office grant W911NF-15-1-0577 and the National Science Foundation grant IIS-1817046.

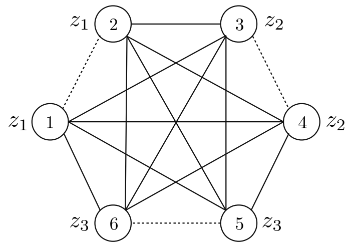


Figure 1: Network W^{KC} for $n = 3$; absent edges C are displayed dashed. Node states $z \otimes \mathbb{1}_2$, used in the reduction, are displayed next to the nodes.

Let us show what DIVER transforms into under the proposed input of (3). First, we notice that, for $b(k, x, \widetilde{x}) = \langle \pi, (m+k)x - m\widetilde{x} \rangle$ of (2), the following holds

$$b\left(k, \frac{s\mathbb{1} + m\widetilde{x}}{m+k}, \widetilde{x}\right) = \langle \pi, s\mathbb{1} + m\widetilde{x} - m\widetilde{x} \rangle = s\langle \pi, \mathbb{1} \rangle = s.$$

Then, DIVER's objective (2) under input (3) will look as

$$f'(\widetilde{W}^{KC}) = \left| \left\langle \sum_{(i,j) \in S} (e_i + e_j), \frac{z \otimes \mathbb{1}_2}{2} \right\rangle - s \right| = \left| \sum_{\ell=1}^{n} y_\ell z_\ell - s \right|,$$

where y_ℓ are edge decision variables

$$y_\ell = \begin{cases} 1, & \text{if } (2\ell - 1, 2\ell) \in S, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, solving DIVER via minimizing $f'(\widetilde{W}^{KC})$, we look for a subset of $\{z_\ell\}$ of size k summing up to s, which is exactly what kSSP_{01} is after, so $\text{kSSP}_{01} \propto$ DIVER.

Parts 1) and 4) of the proof together establish SSP $\propto \text{kSSP}_{01} \propto$ DIVER, so DIVER is NP-hard. □

COMBINATORIAL NETWORK DESIGN

These works target network design problems with combinatorial objectives and/or methods. A large share of these works is dedicated to direct information spread optimization in combinatorial opinion dynamics models, in contrast to indirectly optimizing some analytic feature of the network, such as the spectral radius of its adjacency matrix, expected to facilitate or hinder information propagation.

Information Spread: Chaoji et al. [2] look at a problem of maximizing the number of activated users under the Independent Cascade-like opinion dynamics model in an undirected network via edge addition. The authors prove NP-hardness of the problem, apply continuous relaxation to gain submodularity of the objective, and design a greedy cubic-time approximation algorithm for the relaxed problem. Kuhlman et al. [7] focus on general threshold-based propagation models, and address the problem of minimizing the contagion spread via edge deletion in a directed weighted network. The authors prove inapproximability of the problem, and design a spread simulation-based heuristic, that proves to be effective in experiments. The work of Khalil et al. [6] is dedicated to facilitating or hindering the spread of information under Linear Threshold (LT) model via edge addition or deletion in a directed weighted network. The authors design an influence objective function and prove its supermodularity. The latter property used together with LT process sampling results in a linear-time edge selection algorithm.

Shortest Paths and Optimal Flows: Phillips [12] studied the problem of minimizing maximum flow in a network. Each capacitated edge has a destruction cost, and the adversary needs to choose some edges to destroy, constrained by the edge destruction budget. The author proves NP-hardness of the problem, and designs an FPTAS for planar networks. Israeli and Wood [4] study an NP-hard problem of maximizing a single s-t shortest path in a directed network via edge removal, formulated as a mixed-integer program (MIP). Due to the high time complexity of a direct solution of a MIP problem, the authors propose decomposition techniques to accelerate the computation under some assumptions on the edge removal delays. Papagelis et al. [9] address the problem of minimizing the average all-pairs shortest path length in a connected undirected network via edge addition, and propose a greedy algorithm and two heuristics. Their most efficient algorithm has a quadratic time complexity. Ishakian et al. [3] define a general path-counting centrality measure and study a problem of maximizing the centrality of a given node via edge addition in a DAG. The authors use a quadratic-time greedy strategy for picking edges providing the largest marginal increase of the objective. Parotsidis et al. [10] study minimization of the sum of lengths of the shortest paths from a target node to all other nodes by recommending a link to the target node in an undirected network. They prove NP-hardness of the problem and design an efficient approximation algorithm, relying on submodularity of the objective. A related problem of minimizing the maximal shortest path length has been previously addressed by Perumal et al. [11]; another related problem of maximizing the coverage or betweenness centralities is addressed by Medya et al. [8].

PROBABILISTIC EDGE ADDITION

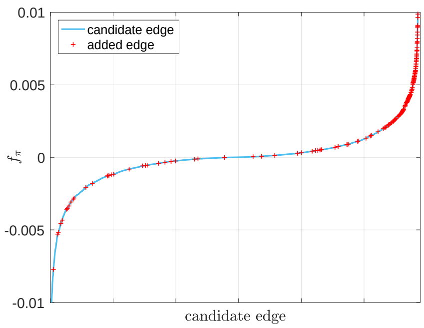
Our theory and algorithms described in the main paper [1] straightforwardly extend to the case of probabilistic edge recommendation. Let us assume candidate edge (r, c) is accepted with probability $\mathbb{P}_{rc} \in [0, 1]$, that can be estimated based on the present links and the history of their creation. (Alternatively, we can choose $\mathbb{P}_{rc} \in \{0, 1\}$ ourselves, solving a problem where only a subset of candidate edges is available for recommendation.) Then, due to linearity of the inner product operation, the probabilistic version $f_\pi^{\mathbb{P}}(r, c)$ of the edge score $f_\pi(r, c) = \langle \pi - \widetilde{\pi}, \widetilde{x} \rangle$ is

$$f_\pi^{\mathbb{P}}(r, c) = \mathbb{P}_{rc}\langle \pi - \widetilde{\pi}, \widetilde{x} \rangle + (1 - \mathbb{P}_{rc})\langle \pi - \pi, \widetilde{x} \rangle = \mathbb{P}_{rc} f_\pi(r, c),$$

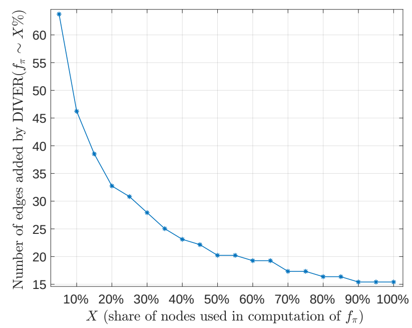
and Algorithm 1 (lines 5, 7) changes in that it computes $f_\pi^{\mathbb{P}}(r, c)$ instead of $f_\pi(r, c)$ using the same procedure from Sec. 5.4. We can further change Algorithm 1 (line 2) to select edge sources not just based on their eigencentrality π_i, but rather based on the value of the product $\pi_i \max_j \mathbb{P}_{ij}$, thereby, avoiding recommending new edges to "celebrity sources" having $\max_j \mathbb{P}_{ij} \approx 0$. Notice that the above proposed alterations do not affect Algorithm 1's time complexity.

DIVER'S EFFECTIVENESS AND EFFICIENCY

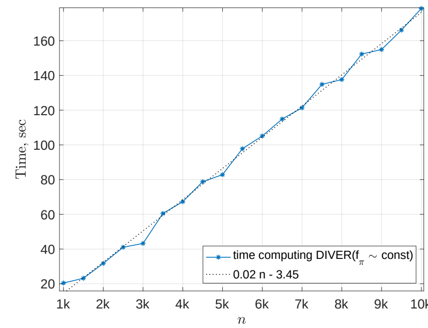
In the main manuscript [1], we observed that performance of DIVER($f_\pi \sim X\%$) deteriorates when X gets small. Here, we illustrate the impact of lower X upon the heuristic's effectiveness. Fig. 2a



(a) Quality of candidate edges added by DIVER($f_\pi \sim 20\%$) **to SF**$(256, -2.5)$**.**



(b) Dependence of DIVER($f_\pi \sim X\%$) **performance on** X **over SF**$(256, -2.5)$**.**



(c) Time for a single evaluation of our heuristic DIVER($f_\pi \sim const$)**, adding** $k = 10$ **edges.**

Figure 2: Effectiveness and efficiency of DIVER($f_\pi \sim X\%$)**.**

shows the quality f_π of each candidate edge of SF$(256, -2.5)$ with $n_{src} = 20$, as well as the quality of edges selected by DIVER($f_\pi \sim 20\%$) over multiple iterations. While due to the approximate computation of f_π, some poor-quality edges are selected—a few of which actually drive the average opinion up—most of the selected candidate edges are high-quality. Fig. 2b shows how the heuristic's performance improves when we throw in more nodes into the computation of f_π. Finally, Fig. 2c shows the actual time it takes to perform a single evaluation of DIVER($f_\pi \sim 100\% \frac{n_{src}}{n}$) adding $k = 10$ edges to $SF(1024, -2.5)$ using an Intel i7-5930K CPU. The heuristic scales linearly, as per Theorem 7. However, if we extrapolate, a single evaluation of DIVER on a $100M$-node network would take around 18 days. Thus, to make the method practical for such massive networks, we can improve its performance even further by considering only some of n nodes as destinations for the candidate edges; for example, we can consider only the destination nodes 2 hops away from the source nodes, increasing the acceptance likelihood of the recommended edges.

DIVER'S ROBUSTNESS TO NETWORK NOISE

We studied what happens to performance of DIVER($f_\pi \sim 100\%$) when we do not know the exact weights of the edges in the network. We run DIVER over $SF(1024, -2.5)$ network when edge weights are known exactly as well as when they deviate from their true values by up to 1% and 10%, respectively. The results are reported in Fig. 3. We observe that DIVER's performance expectedly worsens when we perturb the network, but still remains acceptable.

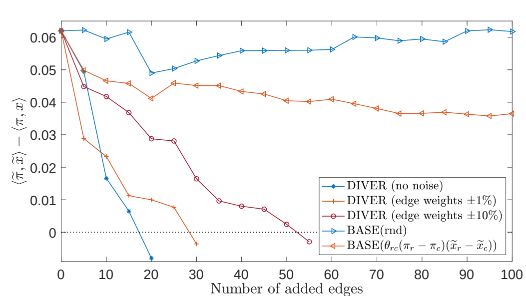


Figure 3: Performance of method DIVER($f_\pi \sim 100\%$) **over** $SF(1024, -2.5)$ **known exactly and non-exactly.**

REFERENCES

[1] Victor Amelkin and Ambuj K. Singh. 2019. Fighting Opinion Control in Social Networks via Link Recommendation. In *Proc. of ACM SIGKDD Conference of Knowledge Discovery and Data Mining (KDD'19)*. ACM, Anchorage, AK, US. https://doi.org/10.1145/3292500.3330960

[2] Vineet Chaoji, Sayan Ranu, Rajeev Rastogi, and Rushi Bhatt. 2012. Recommendations to boost content spread in social networks. In *Proc. of International Conf. on World Wide Web (WWW)*. ACM, 529–538.

[3] Vatche Ishakian, Dóra Erdös, Evimaria Terzi, and Azer Bestavros. 2012. A framework for the evaluation and management of network centrality. In *Proc. of Intern. Conf. on Data Mining (SDM)*. SIAM, 427–438.

[4] Eitan Israeli and R Kevin Wood. 2002. Shortest-path network interdiction. *Networks* 40, 2 (2002), 97–111.

[5] Richard M Karp. 1972. Reducibility among combinatorial problems. In *Complexity of Computer Computations*. Springer, 85–103.

[6] Elias Boutros Khalil, Bistra Dilkina, and Le Song. 2014. Scalable diffusion-aware optimization of network topology. In *Proc. of Intern. Conf. on Knowledge Discovery and Data Mining (KDD)*. ACM, 1226–1235.

[7] Chris J Kuhlman, Gaurav Tuli, Samarth Swarup, Madhav V Marathe, and SS Ravi. 2013. Blocking simple and complex contagion by edge removal. In *Proc. of Intern. Conf. on Data Mining (ICDM)*. IEEE, 399–408.

[8] Sourav Medya, Arlei Silva, Ambuj Singh, Prithwish Basu, and Ananthram Swami. 2018. Group centrality maximization via network design. In *Proc. of Intern. Conf. on Data Mining (SDM)*. SIAM, 126–134.

[9] Manos Papagelis, Francesco Bonchi, and Aristides Gionis. 2011. Suggesting ghost edges for a smaller world. In *Proc. of Intern. Conf. on Information and Knowledge Management (CIKM)*. ACM, 2305–2308.

[10] Nikos Parotsidis, Evaggelia Pitoura, and Panayiotis Tsaparas. 2016. Centrality-aware link recommendations. In *Proc. of Intern. Conf. on Web Search and Data Mining (WSDM)*. ACM, 503–512.

[11] Senni Perumal, Prithwish Basu, and Ziyu Guan. 2013. Minimizing eccentricity in composite networks via constrained edge additions. In *Proc. of Military Communications Conference (MILCOM)*. IEEE, 1894–1899.

[12] Cynthia A Phillips. 1993. The network inhibition problem. In *Proc. of Symposium on Theory of Computing (STOC)*. ACM, 776–785.