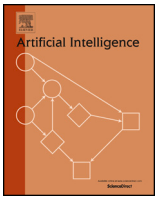
Artificial Intelligence 335 (2024) 104173

Contents lists available at ScienceDirect

Artificial Intelligence

journal homepage: www.elsevier.com/locate/artint



Adversarial analysis of similarity-based sign prediction

Michał T. Godziszewski [a,b,*], Marcin Waniek [c,d], Yulin Zhu [e], Kai Zhou [e], Talal Rahwan [d], Tomasz P. Michalak [c,a]

[a] IDEAS NCBiR, Poland
[b] University of Łodz, Faculty of Mathematics and Computer Science, Poland
[c] University of Warsaw, Faculty of Mathematics, Informatics, and Mechanics, Poland
[d] New York University Abu Dhabi, United Arab Emirates
[e] The Hong Kong Polytechnic University, Department of Computing, Hong Kong

ARTICLE INFO

Keywords:
Signed networks
Adversarial sign prediction

ABSTRACT

Adversarial social network analysis explores how social links can be altered or otherwise manipulated to hinder unwanted information collection. To date, however, problems of this kind have not been studied in the context of signed networks in which links have positive and negative labels. Such formalism is often used to model social networks with positive links indicating friendship or support and negative links indicating antagonism or opposition.

In this work, we present a computational analysis of the problem of attacking sign prediction in signed networks, whereby the aim of the attacker (a network member) is to hide from the defender (an analyst) the signs of a target set of links by removing the signs of some other, non-target, links. While the problem turns out to be NP-hard if either local or global similarity measures are used for sign prediction, we provide a number of positive computational results, including an FPT-algorithm for eliminating common signed neighborhood and heuristic algorithms for evading local similarity-based link prediction in signed networks.

1. Introduction

Increasingly ubiquitous threats to the safety of our private information have fueled interest in the rapidly growing field of adversarial social network analysis. In particular, works from this body of literature study the problem of rewiring or otherwise manipulating the structure of a social network in order to hinder unwanted information gathering. There already exist solutions that can protect our sensitive information from a wide variety of social network analysis tools. Examples of such tools include centrality measures [14,3,49], link prediction algorithms [58,57,18], community detection algorithms [44,20,7], as well as deep learning-based techniques [10]. However, all these existing privacy-protection solutions are only able to process unsigned networks.

There exists a growing literature on signed networks, i.e., networks in which links are labeled with either plus or minus signs [31,32,42]. Such networks can be used to model positive and negative relations between the nodes. For example, the positive links in a social network might indicate friendship or support, while the negative links could represent antagonism or opposition. A vital problem specific to signed networks is sign prediction. It involves determining whether a particular link (the sign of which is yet unknown) has a positive or negative label [31,1,12,24]. Sign prediction methods can be used to improve solutions of network analysis

* Corresponding author.
E-mail address: mtgodziszewski@gmail.com (M.T. Godziszewski).

https://doi.org/10.1016/j.artint.2024.104173
Received 27 December 2022; Received in revised form 1 December 2023; Accepted 11 June 2024

Available online 27 June 2024

tasks such as node ranking [40], anomaly detection [27,53], community detection [43], information diffusion [39,33] and sentiment prediction [51]. However, they can also be used to uncover information about the nature of certain relations which some network members may prefer to keep private. So far, there exist no methods that allow individuals to hide from sign prediction. Our work is the first step to fill this gap in the literature.

To motivate the problem of hiding from sign prediction, let us discuss examples of settings where sign prediction methods can be applied, and elaborate on why some entities might have an incentive to mislead such methods. First, consider a trading platform, where users are allowed to rate their business partners, forming a publicly visible who-trusts-who network. Such trust networks can be found in cryptocurrency trading platforms, such as Bitcoin Alpha, as well as many other e-commerce platforms, e.g., Amazon, eBay, and Flickr. Before conducting a transaction with someone for the first time, a user of such a platform might apply a sign prediction algorithm to evaluate whether their potential partner is trustworthy, i.e., whether the link between them is going to be positive. Hence, a malevolent user of such a platform has incentive to strategically manipulate the signs in the network (e.g., by adding fraudulent reviews) in order to appear more credible. Second, adjusting political campaign strategies to the aggregated sentiments of the tweets related to the election is an increasingly popular practice [37]. In this context, signed social networks are used in the study of unsupervised sentiment analysis [11]. In this context, sign prediction can become a campaign tool allowing the political parties to predict the reaction of members of the general public [41]. Mitigating potential campaign strategies could therefore be achieved by strategically reporting certain tweets, thereby removing them from the set of information available to the organization running the campaign. Third, a slightly lower-stakes setting is Wikipedia, where editors can express positive or negative opinions about each other. These opinions directly influence the decisions of whether or not prolific editors are granted administrator rights [4]. Here, sign prediction can be used to evaluate the trustworthiness of a candidate by predicting the signs of votes that have not been cast. Hence, a group of partisan Wikipedia users might have an incentive to manipulate the voting system in order to promote their candidates by strategically expressing (or refraining from expressing) opinions about certain editors.

Against this background, we present the first computational analysis of attacking sign prediction. Specifically, we formalize five computational problems in which the attacker aims to hide (i.e., keep close to zero) or reverse the signs of a target set of links from the network analyst by removing the signs of some non-target links in the network. In all of the considered problems, we only allow the attacker to remove signs from a selected subset of links. This assumption represents the fact that not all network connections might be amenable to being modified by the attacker. For example, fraudulent Wikipedia admins might only control the opinions expressed by the members of their own group.

We now briefly introduce the considered computational problems. The most general problem is NEUTRALIZING SIGN PREDICTION (NSP), where given a particular signed similarity measure and a target set of links, the goal of the attacker is to remove the signs of a bounded number of links from a specific subset of links so that the absolute value of the similarity measure is below a given threshold for every target link. NEUTRALIZING SIGNED COMMON NEIGHBORHOOD (NSCN) is a variant of NSP restricted to the case where the similarity measure in question is the Signed Common Neighborhood. We distinguish this sub-problem since the study of its specific features in the case when the threshold is set to zero is particularly interesting. In the problem of ELIMINATING SIGNED COMMON NEIGHBORHOOD (ESCN), the goal of the attacker is to remove signs of a bounded number of links so that not only the value of the signed common neighborhood measure for each pair in the target set is equal to zero, but also that for any such pair, there are no signed common neighbors (i.e., nodes connected to both nodes by signed links). Furthermore, we define the problem of NEUTRALIZING TOTAL SIGN PREDICTION (NTSP), where the goal of the attacker is to remove the signs of a bounded number of links so that the sum of absolute values of a given similarity measure for all links in the target set is below the threshold. The last problem we introduce is of REVERSING SIGN PREDICTION (RSP), where the attacker's goal is to delete signs of a bounded number of links in the graph so that the signs of the values of similarity measures of links from the target set are reversed after the removal.

We prove that all of the above problems are NP-hard for the Signed Common Neighborhood (SCN), Signed Jaccard (SJ), and Signed Katz (SK) similarity measures in their unrestricted versions. We also demonstrate that the subproblems of ESCN and NSCN, where the target set of links is derived as the set of all links between distinguished, *important* elements, are P-time computable. We then analyze the parameterized complexity, and demonstrate that a restricted version of the NSCN and ESCN are fixed-parameter tractable (FPT) w.r.t. k which is the budget of the attacker, i.e., the number of links whose signs can be removed by the attacker. Nevertheless, the general version of NSCN turns out to be W[2]-hard for the SCN and SJ measures. We also show that NTSP is W[1]-hard for the same two measures w.r.t. k. We further prove that all the problems apart from RSP are fixed-parameter tractable with respect to the size of the target set. We thus discover an intriguing fact that when the problems are parameterized by the size of the target set, they become tractable, whereas when parameterized by the attacker's budget, they might remain intractable (in the sense of parameterized complexity). Table 1 presents the overview of our computational results.

Given the above computational results, we propose heuristic algorithms for evading similarity-based sign prediction. The first one focuses on eliminating signed common neighborhood and it aims at keeping the similarity measures of all the target links as close to zero as possible. The second one attempts to reverse the signs of the target links. We evaluate both heuristics on real and synthetic datasets and demonstrate their effectiveness. We also introduce an alternative heuristic for both neutralizing sign prediction and reversing sign prediction problems, namely the Tally heuristic. While the first two heuristics are focused on the positive contribution of the removal of sign from the network's link, i.e., how said removal brings the value of signed common neighborhood measure of a target link closer to the desired value, the Tally heuristic takes into consideration the negative contributions as well.

The complexity results we obtain can be useful from a machine learning perspective; the formulations of the problems we consider are conceptually similar to the general class of decision-time attacks on classifiers in adversarial machine learning, and can be utilized while constructing algorithmic solutions against adversarial agents. The hardness results demonstrate a certain degree of safety with

Table 1
A summary of our hardness results. The parameter k denotes the number of signs that the attacker may remove from the observed network; $|H|$ denotes the size of the target set of links; r denotes the threshold below which the attacker wants to set the absolute values of the SCN measures.

Complexity	Parameterized complexity	
	w.r.t. k	w.r.t. $\|H\|$
NEUTRALIZING SIGN PREDICTION (NSP):		
for SCN and SJ: NP-hard (Theorem 1)	W[2]-hard (Corollary 5)	FPT (Corollary 8)
for SK: NP-hard (Theorem 3)	Open problem	Open problem
NEUTRALIZING SIGNED COMMON NEIGHBORHOOD (NSCN):		
NP-hard (Corollary 2)	W[2]-hard (Theorem 8)	FPT (Theorem 7)
ELIMINATING SIGNED COMMON NEIGHBORHOOD (ESCN):		
NP-hard (Corollary 1)	FPT (Theorem 5)	FPT (Corollary 6)
$\exists U\ H = U \times U$: P-time computable (Theorem 4)	—	—
NEUTRALIZING TOTAL SIGN PREDICTION (NTSP):		
NP-hard (Corollary 3)	W[1]-hard (Theorem 7)	FPT (Theorem 10)
REVERSING SIGN PREDICTION (RSP):		
NP-hard (Theorem 2)	Open problem	Open problem

respect to potential attacks in their most general unconstrained form, but positive algorithmic solutions for the attacker in the form of our heuristics identify potential sources of an attack aimed at any given learned likelihood function.

The remainder of this article is organized as follows. We begin with a short overview of related literature in Section 2. Next, in Section 3 we present local and global similarity measures for signed graphs that had been introduced in the literature and that we study in this work. In particular, we briefly discuss how they relate to their unsigned counterparts and show the motivations behind them, especially with respect to the so-called balance theory. In Section 4 we formalize the computational problems that are the main subject of our complexity analysis in Section 5. Furthermore, in Section 6, we introduce four different heuristics to tackle the problems in question, and in Section 7 we evaluate them experimentally. Conclusions follow in Section 8.

2. Related work

Various evasion techniques against a number of social network analysis tools have been investigated in the literature, with the typical goal being to protect privacy. In particular, closely cooperating groups of agents might want to avoid being identified by community detection algorithms [44]. This is especially important given that these algorithms can be used to accurately infer even undisclosed information about social media users, including their sexual orientation or other sensitive characteristics [38,36].

Furthermore, a number of papers analyzed manipulating centrality measures in social networks [13,44]. For instance, the problem of strategically decreasing the value of a node's centrality by modifying a fixed number of relationships was shown to be computationally intractable for the fundamental centrality measures apart from the degree centrality [44]. Interestingly, a simple heuristic turns out to be surprisingly effective in this respect. Another work presented a set of intuitive axioms that characterizes a measure of manipulability of centrality measures [49]. The problem of evading centralities was also analyzed for non-standard network models, such as multilayer networks [46]. Dey and Medya [17] analyzed the problem of hiding network leaders from the core centrality; they also presented a deeper theoretical analysis of the approximation version of the problem of evading degree centrality.

Not only may social network users want to evade centrality measures, but they may also want to evade link prediction algorithms, to avoid exposing their private relationships [55,45,58]. Other types of social network analysis tools for which hiding or evasion techniques were developed include source detection algorithms [47] and node similarity measures [18]. In the latter work, the authors analyze the problem of evading link prediction by removing links in simple networks. They demonstrate that manipulating node similarity measures is often NP-hard, and provide further parameterized complexity results. Our work can be seen as a generalization of their results into the context of signed networks.

We also mention some approaches to sign-prediction that are not based on similarity measures. In particular, Yuan et al. [56] introduce a sign prediction method for negative signs that analyses three categories of features: (i) nodes' features, (ii) triads' features, and (iii) users' similarity features. These three categories are then merged via a logistic regression model. Furthermore, Huang et al. [25] use signed networks to develop a probabilistic framework for modeling trust prediction based on non-classical logic, using two different approaches: a structural balance model based on social triangles, and a social status model based on a consistent status hierarchy. Moreover, Yang et al. [54] study sign prediction via unsupervised and semisupervised algorithms, and demonstrate the possibility of inferring signed social ties with good accuracy solely based on users' behavior of decision making, allowing for turning an unsigned acquaintance network (e.g., Facebook or Myspace) into a signed trust-distrust network.

Further in this direction, Guha et al. [23] develop a framework of trust propagation schemes by extending mathematical approaches to the propagation of trust to the case in which users may also express distrust, by introducing the so-called distrust matrices, and incorporating them into algorithms predicting trust or distrust.

Our approach is different, since we investigate models of sign-prediction based on similarity measures, such as the ones used by Derr et al. [16], who introduce numerous, both local and global, relevance measurements for signed social networks, and investigate the connection between these signed relevance measurements, balance theory, and signed network properties.

3. Background

Let graph $G = (V, E)$ represent an (unsigned) social network, where V is the set of nodes and E is the set of links. Let us denote by $N(v)$ the set of neighbors of $v \in V$, i.e., $N(v) = \{w \in V : \{v, w\} \in E\}$, and by $d(v)$ the degree of $v \in V$, i.e., $d(v) = |N(v)|$. A *similarity measure* is a function that assigns to any pair of nodes a real value that reflects the similarity between them. Two main types of similarity measures can be distinguished: *local* and *global*. The former ones focus on the direct neighborhood of the nodes for which similarity is computed. For example:

- **Common Neighborhood (**CN**):**

$$\mathrm{CN}(u, v) = |N(u) \cap N(v)|$$

- **Jaccard (J):**

$$\mathrm{J}(u, v) = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}$$

- **Preferential Attachment (**PA**):**

$$\mathrm{PA}(u, v) = d(u) \times d(v)$$

In contrast, global similarity measures take into account the entire graph. An example is the Katz measure. To introduce it, let us first denote by $A = (a_{ij})_{v_i, v_j \in V}$ the adjacency matrix of a given graph, where $a_{ij} = 1$ if $\{u, v\} \in E$, and $a_{ij} = 0$ if $\{u, v\} \notin E$. Let $paths^l_{u,v}$ denote the set of paths of length l between u and v in the graph G, and suppose $\beta < 1$ is a (small) real number. Then, the Katz Index K of the pair (u, v) is defined as:

- **Katz:**

$$\mathrm{K}(u, v) = \sum_{l=1}^{\infty} \beta^l \cdot |paths^l_{u,v}| = \sum_{l=1}^{\infty} \beta^l \cdot A^l.$$

In this paper, we focus on the counterparts of the above measures, defined for signed social networks that are represented by graphs in which links additionally have labels, either positive or negative. Formally, a *signed social network* is a graph $G = (V, E, \sigma)$, where $\sigma : E \to \{+, -\}$ is a *sign function* on the links. Furthermore, let us denote by $N_+(v)$ and $N_-(v)$ the positive and negative neighborhood of a node $v \in V$, respectively, i.e.:

$$N_+(v) = \{w \in V : \{v, w\} \in E \ \&\ \sigma(\{v, w\}) = +\}, \text{ and}$$

$$N_-(v) = \{w \in V : \{v, w\} \in E \ \&\ \sigma(\{v, w\}) = -\}.$$

Given these definitions, we are now ready to introduce the notion of **similar common neighborhood** of $u, v \in V$, denoted by $c_s(u, v)$, i.e., the set of nodes adjacent to both u and v connected to them via links of the same signs. Formally:

$$c_s(u, v) = (N_+(u) \cap N_+(v)) \cup (N_-(u) \cap N_-(v)).$$

Analogously, let $c_d(u, v)$ denote the set of **dissimilar common neighborhoods** of $u, v \in V$, i.e., the set of nodes adjacent to both u and v connected to them via links of the opposite signs. Formally:

$$c_d(u, v) = (N_+(u) \cap N_-(v)) \cup (N_-(u) \cap N_+(v)).$$

Let d_v denote the degree of the node v, i.e., the number of v's neighbors. We will also use $d_+(v)$ and $d_-(v)$ to denote the number of neighbors of v with which this node has positive and negative connection, respectively, i.e., $d_+(v) = |N_+(v)|$ and $d_-(v) = |N_-(v)|$. Let the positive and negative preferential attachments between u and v be denoted by $\mathrm{PA}_+(u, v) = d_+(u) \cdot d_+(v)$ and $\big(\mathrm{PA}_-(u, v) = d_-(u) \cdot d_-(v)\big)$, respectively.

While there are many local similarity measures for non-signed networks,[1] only recently some of their counterparts for signed networks have been analyzed in the literature [16,9]. These are:

[1] Additional examples of measures other than those already introduced in this section can be found, e.g., in the work by Linyuan and Tao [34].

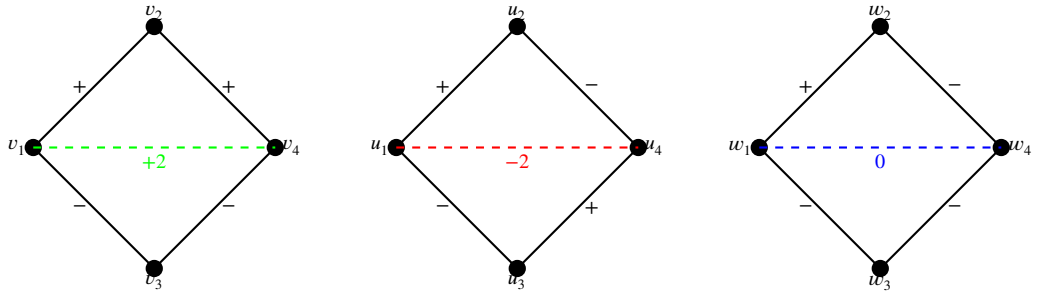


Fig. 1. Three sample signed networks of four nodes. The weights of the colored dashed links are the SCN scores of the pairs of nodes $\{u_1, u_4\}$, $\{v_1, v_4\}$, and $\{w_1, w_4\}$. In contrast, the unsigned counterpart of SCN, i.e. the CN measure, outputs a score equal to 2 in all the three cases. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

- **Signed Common Neighborhood** (SCN, see [8], [16]):

$$\begin{aligned}\mathrm{SCN}(u,v) &= |c_s(v,u)| - |c_d(v,u)| \\ &= \big(|N_+(u) \cap N_+(v)| + |N_-(u) \cap N_-(v)|\big) + \\ &- \big(|N_+(u) \cap N_-(v)| + |N_-(u) \cap N_+(v)|\big).\end{aligned}$$

- **Signed Jaccard** (SJ, see [16]):

$$\mathrm{SJ}(u,v) = \frac{|c_s(u,v)| - |c_d(u,v)|}{|N(u)_+ \cup N(u)_- \cup N_+(v) \cup N_-(u)|}.$$

- **Signed Preferential Attachment** (SPA, see [16]):

$$\mathrm{SPA}(u,v) = sgn(\mathrm{PA}_+(u,v) - \mathrm{PA}_-(u,v))\, f(\mathrm{PA}_+(u,v), \mathrm{PA}_-(u,v)),$$

  where f is any function aggregating PA_+ and PA_- into a measure of relevance strength. It has been established empirically in [16] that a function that works best as f is $\max(\mathrm{PA}_+(u,v), \mathrm{PA}_-(u,v))$.

To illustrate the difference between the above three measures for signed networks and their counterparts for unsigned networks, let us consider the graphs in Fig. 1. Let us first disregard signs of links as standard similarity measures do. Then, in all the three graphs, pairs $\{v_1, v_4\}$, $\{u_1, u_4\}$, and $\{w_1, w_4\}$ have 2 common neighbors, and this would be the value of the common neighborhood similarity measure for unsigned graphs. However, when we take into account the signs of links, the values of SCN for these pairs are 2, −2, and 0, respectively. The value of $\mathrm{SJ}(\{u,v\})$ is equal to the quotient of $\mathrm{SCN}(\{u,v\})$ by the cardinality of the sum of all signed neighborhoods of u and v.

In Fig. 3 and Table 2 we demonstrate the combinatorial difference between the tasks of manipulating nodes' similarity by removing edges or their signs in the different types of networks (in the realm of signed networks, the task can be much more complex or require a higher budget of operations).

Observe that the SJ and SCN measures are monotone with respect to each other. However, the Signed Preferential Attachment (SPA) scores can differ significantly from the SCN and SJ scores of the same pairs. This is due to the fact that the sign of the SPA score of a given pair $\{x, y\}$ of nodes represents the difference in the products of their positive and negative PA scores. This means that the sign of the SPA score of $\{x, y\}$ is positive (negative) if the product of the positive neighborhoods of x and y is strictly greater (smaller) than the product of their negative neighborhoods, and then, if positive, the absolute value of the $\mathrm{SPA}(x, y)$ measures greater of those products. On the other hand, if $\mathrm{SPA}(x, y)$ is equal to zero, this means that the products of their positive and negative neighborhoods are identical, but not necessarily equal to 0. Therefore, following [16] we can interpret SPA as giving justice to the intuition that the *rich get richer*—a high positive value of SPA means there are *many* positive neighbors of both x and y relative to the number of their negative neighbors, while a high negative value of SPA means that there are many negative neighbors of x and y, relative to the number of their positive neighbors. The difference between SPA and other local signed measures is presented in Fig. 2. The figure also illustrates how the signed counterparts of the similarity measures (whatever they should intuitively measure) differ significantly from their unsigned counterparts.

Global similarity Let $A = (a_{ij})_{v_i, v_j \in V}$ denote the adjacency matrix of a given signed graph, where $a_{ij} = 1$ if $\sigma(\{v_i, v_j) = +$, $a_{ij} = 0$ if $\{v_i, v_j\} \notin E$, and $a_{ij} = -1$ if $\sigma(\{v_i, v_j\}) = -$. Furthermore, let A^+ (A^-) denote the adjacency matrix of positive (negative) links in the graph, i.e. $a_{ij} = 1$ if $\sigma(\{v_i, v_j\}) = +$ (if $\sigma(\{v_i, v_j\}) = -$) and 0 otherwise.

According to balance theory [5], a path between two nodes in the graph is balanced if it contains an even number of negative links and it is unbalanced otherwise. For a given signed graph G, we then can, in accordance with balance theory, define inductively the following matrices B_l and U_l, the entries of which are the numbers of balanced and unbalanced paths of length l, respectively. Formally:

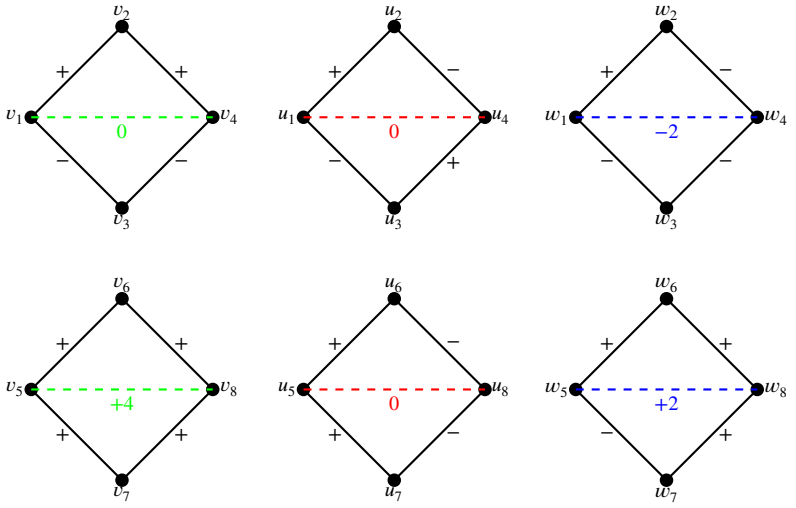


Fig. 2. The SPA scores of the pairs of nodes $\{v_1, v_4\}$, $\{v_5, v_8\}$, $\{u_1, u_4\}$, $\{u_5, u_8\}$, $\{w_1, w_4\}$, and $\{w_5, w_8\}$. Note that the unsigned counterpart of SPA—the PA metric—outputs score equal to 4 in all six cases. Furthermore, the SPA scores are not monotone with respect to the SCN or SJ scores, since they measure a different kind of signed similarity.

$$B_l = (b^l_{ij})_{v_i, v_j \in V} \qquad (U_l = (u^l_{ij})_{v_i, v_j \in V}),$$

where b^l_{ij} (u^l_{ij}) is the number of paths p between v_i and v_j such that $|p| = l$ and the number of negative links in p is even (odd), defined as:

$$B_1 := A^+; \quad U_1 := A^-;$$

$$B_{l+1} := B_l \cdot A^+ + U_l \cdot A^-; \quad U_{l+1} := U_l \cdot A^+ + B_l \cdot A^-.$$

To see how this definition is motivated by balance theory, consider the closed triangles in G. Take a triangle $\{u, v, w\} \in V$ with links between all given nodes, for example. Following the intuitive idea that *the friend of my friend is my friend, the enemy of my enemy is my friend*, and *the friend of my enemy is my enemy*, and *the enemy of my friend is my enemy*, if $\sigma(\{u, v\}) = \sigma(\{v, w\})$, then for $\{u, v, w\}$ to be balanced we need $\sigma(\{u, w\}) = +$. Analogously, if $\{u, v, w\}$ is to be balanced, and $\sigma(\{u, v\}) \neq \sigma(\{v, w\})$, then $\sigma(\{u, w\}) = -$. In both cases, there is a negative number of negative links in the triangle.

Now the idea of the inductive definition above is that adding one positive link (i.e., from A^+) to a balanced (i.e., with an even number of negative links) path of length l (i.e., from B_l), or adding one negative link (i.e., from A^-) to an unbalanced path of length l (i.e., from U_l) results in a balanced path of length $l + 1$ (i.e., from B_{l+1}). The inductive condition for U_{l+1} has an analogous explanation.

Having defined B_l and U_l, we can get a signed version of the Katz global similarity measure for a given signed connected graph[2]:

- **Signed Katz** (SK; see [16]):

$$\mathrm{SK}(v_i, v_j) = \sum_{l=1}^{\infty} \beta^l (b^l_{ij} - u^l_{ij}),$$

where $\beta \in (0, 1)$ is a parameter that gives an exponential decay on the count of paths with their length increasing.

4. Attack model

Let $G^* = (V, E, \sigma(E))$ be a signed graph, and let $H \subseteq (V \times V) \setminus E$ be the set of pairs of nodes not in E. Assume that the pairs of nodes $\{u, w\} \in H$ are actually linked, i.e., we consider the graph $G = (V, E \cup H, \sigma(E))$. Now, the links in H are the attacker's target. Specifically, the aim of the attacker is to make the signs in H more difficult to infer by the defender. To do so, the attacker is allowed to remove the signs of no more than k links in E. That is, k can be interpreted as the attacker's budget. The defender infers the signs in H using a signed similarity measure, which takes as input the entire network G^*. This assumption is quite realistic in settings where the defender is the network operator, e.g., the company behind a particular social media platform, or the owner of the trading platform being used by the attacker.

[2] Here, by the graph being "signed connected", we mean that between any two nodes v and u, there is a signed path.

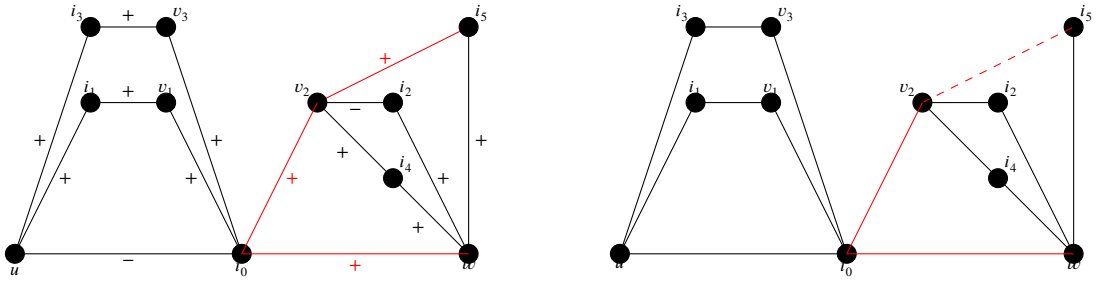


Fig. 3. An instance of ESCN (and thus, NSCN) and its unsigned counterpart. Let the elements of the target set in both cases be the pairs $\{u,w\}$, $\{u,v_1\}$, $\{u,v_3\}$, $\{w,v_2\}$, $\{v_1,v_2\}$, and $\{v_2,v_3\}$. Notice that by removing the signs of the links $\{i_0,w\}$, $\{i_0,v_2\}$, and $\{v_2,i_5\}$ we can make the updated SCN′ scores of all the pairs in the target set equal to 0. Simultaneously, if we consider the unsigned counterpart of the network (on the right), then we cannot manipulate the CN scores of the pairs of nodes in the target set by deleting the corresponding links. Moreover, the minimal number of links that would need to be removed so that for each pair $\{x,y\}$ in the target set $\mathrm{CN}'(x,y)=0$ is 7. This example signifies the difference between the task of manipulating nodes similarity by deleting the (signs) of links in the signed and unsigned networks, demonstrating that the former might be less costly (in terms of the number of links (signs of) that need to be removed), but combinatorially more complex.

Table 2
The values of Common Neighborhood (CN) and Signed Common Neighborhood (SCN) for the pairs of links in the target set of the network from Fig. 3 and the values of these scores updated to CN′ and SCN′ after deleting the (signs of) links colored in red in the same figure. The signed network is a positive instance of NSCN with the goal $r=0$ and budget $k=3$, while its unsigned counterpart is a negative instance of NCN (and a fortiori, ECN) with the same goal and budget.

Link	CN	CN′	SCN	SCN′
$\{u,w\}$	1	1	-1	0
$\{u,v_1\}$	2	2	0	0
$\{u,v_3\}$	2	2	0	0
$\{w,v_2\}$	4	1	2	0
$\{v_1,v_2\}$	1	0	1	0
$\{v_2,v_3\}$	1	0	1	0

From the point of view of the defender, the above setting has three alternative, slightly different, interpretations: (i) the defender knows about the existence of the links in H, but not their signs (e.g., because the links were formed only recently); (ii) the defender has incomplete knowledge about the network structure, and is unsure about both the existence and the signs of edges in H; (iii) the links in H do not exist yet, but the defender expects their future formation, and wishes to predict their signs. All of these interpretations turn out to be mathematically equivalent, as the sign prediction methods considered in this work only take into consideration the edges with known signs. Hence, we will not distinguish between them in the formulation of our computational problems. Developing similarity measures that take into consideration unsigned links of the signed network is a potential direction of future work.

Similarly, the actions performed by the attacker, i.e., the removal of signs from some links, also can be interpreted in two different ways: (i) obfuscating whether the signs are positive or negative (e.g., zeroing the trust ratings between users in a trading platform scenario); (ii) removing not only the signs but also the links themselves, (i.e., removing the tweets in a political campaign scenario). Again, the two interpretations are mathematically equivalent, for the same aforementioned reason.

We formalize our computational problems as follows:

Problem 1 *(NEUTRALIZING SIGN PREDICTION (*NSP*)).* Given a signed graph $G=(V,E\cup H,\sigma(E))$, where $H\subseteq(V\times V)\setminus E$ is the attacker's target set of links, a subset $D\subseteq E$ of links the signs of which can be deleted, an integer $k\leq|D|$ denoting the budget of the attacker, i.e., the maximum number of signs which can be deleted, and a non-negative real number r, decide if there exists a subset $C\subseteq D$ such that $|C|\leq k$ and such that for all $\{u,v\}\in H$ it holds that

$$|sim'(u,v)|\leq r,$$

for a fixed similarity measure $sim: V\times V\to\mathbb{R}$ (observe that sim can take negative values), where $sim'(u,v)$ denotes the value of similarity sim between u and v in graph $G'=(V,E\cup H,\sigma(E\setminus C))$, where the signs of links have been removed. An instance of this problem is a tuple (G,H,D,k,r).

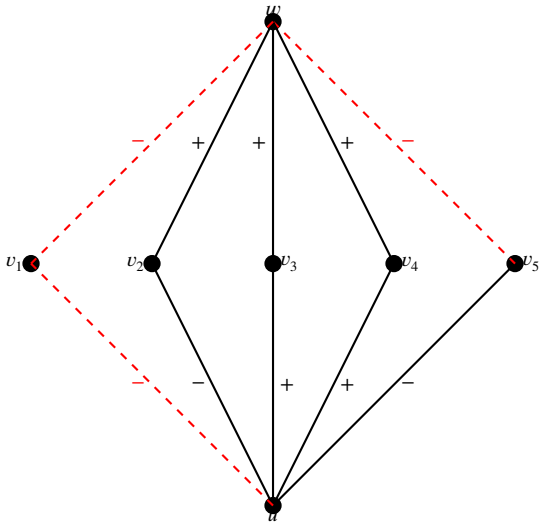


Fig. 4. An instance of NEUTRALIZING TOTAL SIGN PREDICTION (NTSP). Let the target set be $H = \{\{v_i, v_j\} : 1 \leq i < j \leq 5\}$. The sum of absolute values of the SCN scores is $S := \sum_{\{v_i,v_j\}\in H} |\mathrm{SCN}(v_i, v_j)| = 12$. If we delete the signs of $\{v_1, u\}$, $\{v_1, w\}$ and $\{v_5, w\}$, then the value of S drops to 4. Therefore, the network in the figure is a positive instance of NTSP with $k = 3$ and $r = 4$.

Throughout this article, the measures of similarity marked with the prime sign (e.g., sim' above) refer to the measures computed in the network after the hiding process is completed (i.e., after the removal of signs from the links in C).

We distinguish two interesting subproblems of NSP, strictly related to the (Signed) Common Neighborhood measure and the (Signed) Jaccard measure. First of all, we might ask if it is possible to neutralize the SCN scores of all pairs in the target set, meaning that after removing signs of at most k links, the absolute values of the SCN scores of these pairs would all be smaller than a given non-negative integer r.

Problem 2 *(NEUTRALIZING SIGNED COMMON NEIGHBORHOOD (*NSCN*)).* NSCN is a variant of NSP in which instead of asking if the absolute values of a given similarity measure sim can be reduced below a threshold r for all pairs $\{u, v\} \in H$, we ask if there exists a subset $C \subseteq D$ such that $|C| \leq k$ and such that for all $\{u, v\} \in H$ it holds that

$$\big||c_s(u, v)| - |c_d(u, v)|\big| \leq r.$$

Problem 3 *(ELIMINATING SIGNED COMMON NEIGHBORHOOD (*ESCN*)).* ESCN is a variant of NSP in which we ask if there exists a subset $C \subseteq D$ such that $|C| \leq k$ and such that for all $\{u, v\} \in H$ it holds that

$$c_s(u, v) \cup c_d(u, v) = \emptyset.$$

An instance of this problem is a tuple (G, H, D, k).

Note that for the SCN or SJ similarity measures, a positive solution to ESCN is also a positive solution to both NSCN and NSP with $r = 0$. It is also worth noting that this problem is oblivious to the signs of the edges, and thus is an extension of the ELIMINATING SIMILARITY problem by Dey and Medya [18] to the signed networks.

We also study an interesting version of NSP that takes into consideration the sum of moduli of the similarity scores of all the pairs in the target set:

Problem 4 *(NEUTRALIZING TOTAL SIGN PREDICTION (*NTSP*)).* Given $G = (V, E \cup H, \sigma(E))$, $H \subseteq (V \times V) \setminus E$, $D \subseteq E$, k, and r as in Problem 1, decide if there exists a subset $C \subseteq D$ such that $|C| \leq k$ and such that:

$$\sum_{\{u,v\}\in H} |sim'(u, v)| \leq r.$$

An instance of this problem is a tuple (G, H, D, k, r).

Fig. 4 presents an example of an instance of the NTSP problem.

Finally, we also analyze a variant of the problem, where the goal of the attacker is to reverse the sign of the similarity score of the links from the target set rather than neutralize this score. This variant is motivated by balance theory [5] to the extent that one of the ways to hide the signs of the links from the defender is to remove balanced triangles from the observed network.

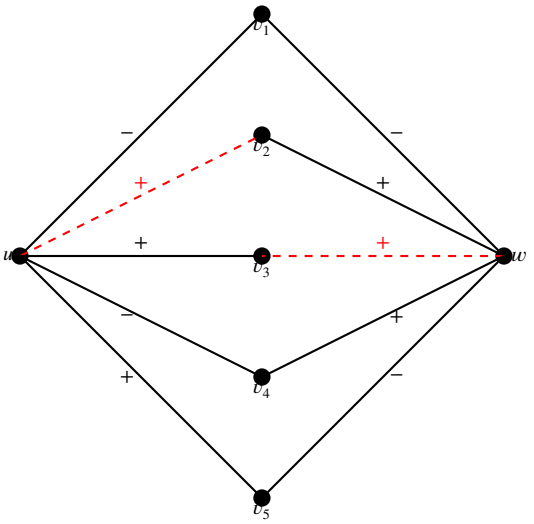


Fig. 5. An instance of REVERSING SIGN PREDICTION (RSP). Let the target set be $H = \{\{u, w\}\}$. The Signed Common Neighborhood of (u, w) is initially equal to $+1$, but if we remove the signs of $\{u, v_2\}$ and $\{w, v_3\}$, then it becomes -1, i.e., it gets reversed. Thus, the network in the figure is a positive instance of RSP with $k = 2$.

Problem 5 *(REVERSING SIGN PREDICTION (*RSP*)).* Given $G = (V, E \cup H, \sigma)$, $H \subseteq (V \times V) \setminus E$, $D \subseteq E$, and k as in Problem 1, decide if there exists a subset $C \subseteq D$ such that $|C| \leq k$ and such that for all $\{u, v\} \in H$ it holds that

$$sgn(sim'(u, v)) = -sgn(sim(u, v)).$$

An instance of this problem is a tuple (G, H, D, k).

Fig. 5 presents an example of an instance of the RSP problem.

5. Complexity analysis

In this section, we first demonstrate that NSP for both local measures and a specific global one is NP-hard, as well as RSP for local measures. Additionally, we show NP-hardness of ESCN and NSCN, even in the most restricted case when $r = 0$. We then demonstrate tractability of ESCN and NSCN under the condition that the target set of links H is induced by a distinguished set of *important* nodes. Further, we analyze the parameterized complexity of the problems above, with respect to parameters k (the budget of the links that the Attacker may remove) and $|H|$ (the size of the target set of links).

5.1. Complexity and NP-hardness

We begin by demonstrating that NSP is NP-hard for local measures, even for the most restricted case, and that NSCN and ESCN are NP-hard as well:

Theorem 1. *Let sim be a local similarity measure such that, for any signed graph $G = (V, E, \sigma)$ and any $u, v \in V$, it holds that:*

$$|c_s(u, v)| - |c_d(u, v)| = 0 \Leftrightarrow sim(u, v) = 0 \tag{1}$$

Then, NSP *is* NP*-hard even if $r = 0$.*

Clearly, both SCN and SJ belong to the local similarity measures that satisfy the condition in the above theorem.

Proof. The technique used in this proof is based on the proof of Theorem 8 from Dey and Medya [18]. We adapt it to the setting with signed networks and sign prediction algorithms.

We prove a stronger result which says that even attacking similarity-based sign prediction by eliminating common signed neighborhood (i.e. making $c_s(u, v) \cup c_d(u, v) = \emptyset$ for all $u, v \in H$) is NP-hard. The theorem trivially follows. We reduce from the VERTEX COVER (VC) problem (see [21]) which is to decide, for a given graph $G = (X, E_X)$ and an integer $k \in \mathbb{N}$, whether there exists a vertex cover of G of size at most k, i.e. a subset $U \subseteq X$ with $|U| \leq k$ such that each link in E_X is incident to some node from U.

Let $I = (G, k)$ be an instance of VC. Assume $|X| = n$ and fix any numbering of the elements in $X = \{x_1, \ldots, x_n\}$. First, we construct a signed graph $(V, E_V \cup H, \sigma(E_V))$, in which the nodes correspond to the **original nodes**. In particular, for each node $x_i \in X$, we construct its *copy* $v_i \in V$, and a single **root node** u that is added to V. The set of links E_V and their signs σ are defined as follows: for each $i \leq n$ we construct a positive link between u and v_i. We will refer to these links as *root links*.

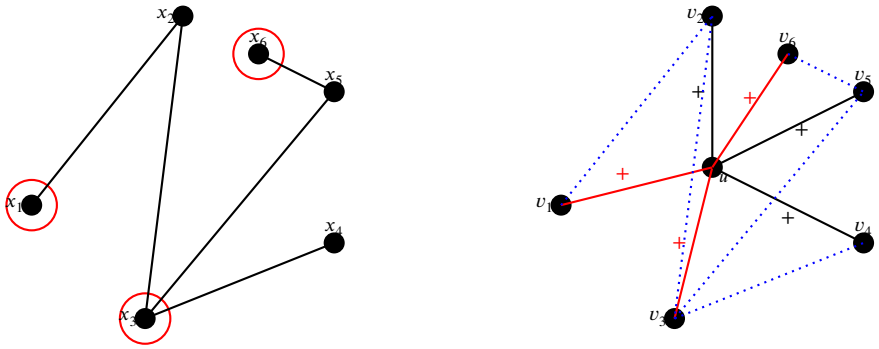


Fig. 6. An illustration of the reduction from VC to NSP in the proof of Theorem 1. The set of nodes $\{x_1, x_3, x_6\}$ forms a vertex cover of the graph $G = (X, E_X)$ on the left. Let us consider now the graph (V, E_V, σ) on the right and let us define the pairs in the target set H as corresponding to the links in E_X, i.e. $\{v_i, v_j\} \in H$ iff $\{x_i, x_j\} \in E_X$. Removing the signs of the links connecting v_1, v_3 and v_6 to u makes the SCN of all the pairs in H equal to 0.

Let us define the pairs in the target set H as corresponding to the links in E_X, i.e. for all $v_i \neq v_j \in V$, we put $\{v_i, v_j\} \in H$ iff $\{x_i, x_j\} \in E_X$. Furthermore, let us set $D := E_V$. Finally, let the number of the links that can have their signs removed in the constructed instance of NSP be equal to k. For an illustration of this reduction, see an example in Fig. 6.

We now need to show that the above reduction is correct for all the measures that satisfy condition (1). To this end, suppose that (G, k) is a "yes" instance of VC. We first prove that this implies that $(V, E_V \cup H, \sigma(E_V))$ is a "yes" instance of NSP. Let $U \subseteq X$ with $U = \{x_{i_1}, \ldots, x_{i_k}\}$ be a vertex cover of G of size k. In such the case, the Attacker removes the positive signs of the links (v_{i_j}, u) for all $j \leq k$. We now claim that for all pairs in the target set, i.e. for each $\{v_i, v_j\} \in H$, the value of $sim(v_i, v_j)$ is equal to 0, since the similar common neighborhood of v_i and v_j is empty, i.e. $c_s(v_i, v_j) = \emptyset$. Indeed if $\{v_i, v_j\} \in H$, then by construction, $\{x_i, x_j\} \in E_X$. By the definition of vertex cover, at least one of the nodes x_i, x_j is in U. Furthermore, observe that before the sign removal it held that $c_s(v_i, v_j) = \{u\}$. Therefore, now we have that $c_s(v_i, v_j) = \emptyset$, and thus $sim'(v_i, v_j) = 0$.

For the other direction, suppose that, after deleting at most k signs of the links in the graph (V, E_V, σ), for each pair $v_i, v_j \in H$ it holds that $sim'(v_i, v_j) = 0$. We will now demonstrate that this implies that there is a vertex cover of the size at most k in the graph $G = (X, E_X)$. Indeed, from the fact that all the links had the positive sign and since $sim'(v_i, v_j) = 0$ holds for all $v_i, v_j \in H$, it follows that $c_s(v_i, v_j) = \emptyset$ after the removal of some signs. Additionally, note that for each pair $v_i, v_j \in H$, their similar common neighborhood before the removal was $c_s(v_i, v_j) = \{u\}$. But these imply that the links, the signs of which are removed, are of the form $\{v, u\}$ for $v \in dom(H)$. Let the links with their signs removed be $\{v_{i_1}, u\}$, $\{v_{i_2}, u\}$, …, $\{v_{i_k}, u\}$. We claim that $U = \{x_{i_1}, \ldots, x_{i_k}\}$ is a vertex cover of G. Indeed, let $\{x_l, x_m\} \in E_X$ be any link from G. Then, by assumption $\{v_l, v_m\} \in H$. Since after the sign removal $c_s(v_l, v_m) = \emptyset$, one of the links $\{v_l, u\}$ or $\{v_m, u\}$ had its sign removed, but by construction this means exactly that either x_l or x_m belongs to U. Since the choice of the link was arbitrary, this ends the proof. □

The result immediately gives us the following corollaries:

Corollary 1. ESCN *is* NP-*hard.*

Corollary 2. NSCN *is* NP-*hard, even when* $r = 0$.

Corollary 3. NTSP *is* NP-*hard, even when* $r = 0$.

A modification of the above reduction also allows us to prove the following:

Theorem 2. *Let* sim *be a local similarity measure such that, for any signed graph* $G = (V, E, \sigma)$ *and any* $u, v \in V$*, it holds that:*

$$sgn(sim(u, v)) = sgn(|c_s(u, v)| - |c_d(u, v)|) \tag{2}$$

Then, RSP *is* NP-*hard.*

Again, both SCN and SJ belong to the local similarity measures that satisfy the condition in the statement of Theorem 2.

Proof. The proof goes by a reduction from VERTEX COVER (VC) very similar to the one used in the proof of Theorem 1. Given a graph $G = (X, E_X)$ and an integer $k \in \mathbb{N}$, we construct a signed graph $G^* = (V, E_V \cup H, \sigma(E_V))$, where the set of nodes V consists of:

- the copies of the original nodes (as in the proof of Theorem 1);
- two additional nodes $\{u_0, u_1\}$ that we call root nodes; and

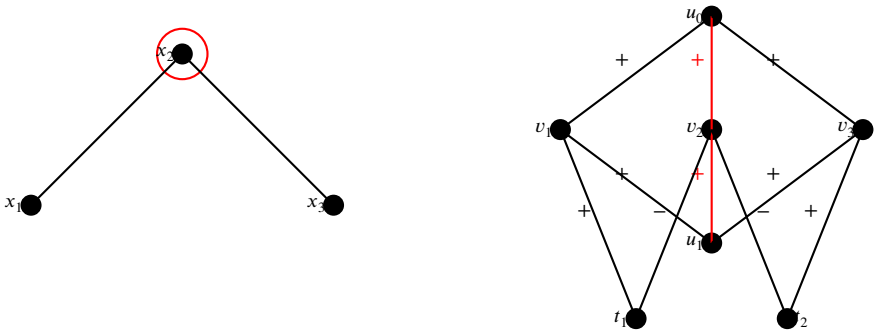


Fig. 7. The illustration of the reduction from VC to RSP in the proof of Theorem 2. The singleton $\{x_2\}$ forms a vertex cover of the graph $G = (X, E_X)$ on the left. In the graph (V, E_V, σ) on the right the target set H is the copy of E_X.

- a set of $m = |E_X|$ additional points $t_1, \ldots, t_m$.

The set of links consists of a family of root links and some additional links. Specifically:

- for each $i \leq n$, we construct a positive root link between v_i and u_0, as well as a positive root link between v_i and u_1;
- for each $l \leq m$ and for each link $e_l = \{x_i, x_j\} \in E_X$, we create two additional links in G^*: a positive link $\{v_i, t_l\}$, and a negative link $\{v_j, t_l\}$, arbitrarily choosing v_i and v_j; and
- as in the proof of 1, we let H to be the copy of E_X, and we put $k^* = 2k$.

For an illustration of this reduction, see an example presented in Fig. 7.

Now, every pair of nodes from H has a positive SCN equal to 1. It is not hard to see that removing a sign from every root link between one of the roots and one of the nodes from the vertex cover of G corresponds to a "yes" instance of RSP. □

As it turns out, computing the solution to NSP for some signed global measures is hard as well—the following negative result holds for Signed Katz:

Theorem 3. *The* NSP *problem for the Signed Katz measure is* NP*-hard, even if the target set H contains only a single link.*

Before getting into the proof let us recall that with respect to the Katz measure we are restricting attention to graphs which are connected (by signed paths). Therefore, if all the paths between given two nodes are of the same type (e.g., all of them are balanced), the Signed Katz measure is by definition minimized for a connected graph when this graph is a path graph with u and w as two end nodes. In such situation our NSP problem is equivalent to the following: given a connected graph G and two nodes u and v, is it possible to remove the signs of k links so that after removal, the graph becomes a signed path with u and v as end-nodes.

Proof. The proof goes by reduction from the HAMILTONIAN CYCLE (HC) problem, where we are asked to decide whether there exists a cycle that visits each node in a given connected graph G.

Given an instance $G = (X, E_X)$ of HC, where $|X| = n$, and $|E_X| = m$, let us construct an instance $J = (V_J, E_V \cup H, \sigma(E_V), H, D, k, r)$ of NSP for the Signed Katz measure, starting with the following signed graph:

- The set of nodes V_J of J consists of
  - two *fresh* nodes u, w,
  - the set $\{v_x : x \in X\}$ of copies of the nodes from X,

  Additionally, define l to be the cardinality of the neighborhood of a randomly chosen arbitrary node $x \in X$, that is $l := |N_G(x)|$. Thus, given an arbitrary $x_0 \in X$, we get

$$V_J := \{v_x : x \in X\} \cup \{u, w\}.$$

- Given these nodes, we create:
  - a negative link $\{v_{x_0}, u\}$,
  - positive links $\{v_{x_0}, v_y\}$ for each $y \in N_G(x_0)$,
  - negative links $\{v_y, w\}$ for each $y \in N_G(x_0)$, and
  - positive links that are copies of the other links in E_X.

Furthermore, we define a subset $D \subseteq E$ (of links the signs of which can be deleted) as follows:

$$D := \{\{v_{x_0}, v_y\} : y \in N_G(x_0))\} \cup \{\{w, v_y\} : y \in N_G(x_0))\} \cup \{\{v_i, v_j\} : \{i, j\} \in E_X\}.$$

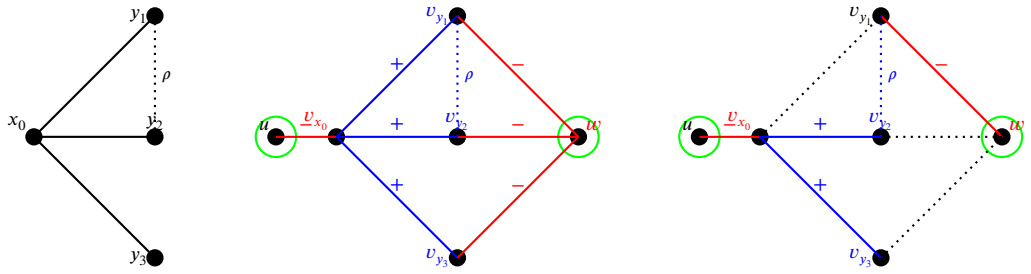


Fig. 8. The illustration of the reduction from HC to NSP for Signed Katz measure. The picture on the left represents a tiny part of an instance of HC. Let x_0 denote the node chosen randomly for the reduction, let the nodes y_1, y_2, and y_3 be its neighbors, and let the dotted line ρ denote some path between y_1 and y_2. That is, for clarity, we skip almost all edges of G, besides the ones in the neighborhood of x_0 and the path ρ. The picture in the middle represents the graph J after the reduction. Two fresh nodes, u, and v (in green circles) have been added, u is negatively connected to v_{x_0}, and w is negatively connected to the nodes from the neighborhood of v_{x_0}. The positive links are denoted in blue, and the negative ones are in red. Since we postulate all the remaining links from G to be positive, the path ρ is a balanced positive path. The right-hand side picture represents the deletion of signs of links in the graph J when G is a positive instance of HC. Assuming y_1 and y_2 are the neighbors of x_0 that are elements of the Hamiltonian Cycle, as well as supposing that ρ is the remaining part of the cycle, we delete the signs of link from v_{x_0} to v_{y_1} and the signs of links connecting all other neighbors of v_{x_0} to w. Apart from that we remove all $m-n$ signs of original links from G that are not included in the fixed Hamiltonian Cycle. The resulting balanced path between u and v of length $n+1$ gives us the value of the Signed Katz score between these nodes equal to β^{n+1}. If the signed graph J is a positive instance of the variant of NSP for Katz measure on connected graphs, then it is possible to remove signs of at most k links so that the resulting network constitutes a path between u and w, and then it is possible to recover the Hamiltonian Cycle in G from that path in J'.

Next, we put $H = \{\{u, w\}\}$ and define $k := l + m - n$. Finally, we complete the reduction by fixing the parameter of the Signed Katz measure β to be any number in the interval $(0, 1)$, and we define $r := \beta^{n+1}$. The reduction is illustrated in the Fig. 8.

We observe that despite the fact we postulated that $\{v_x, u\}$ as well as all $\{v_y, w\}$ are negative, the paths between u and w are still balanced.

Now, assume that G is a "yes" instance of HC. Without loss of generality let the links $\{v_{x_0}, v_{y_1}\}$, and $\{v_{y_2}, v_{x_0}\}$ be elements of a fixed Hamiltonian cycle P from G. Let us remove the sign of $\{v_{y_1}, v_x\}$ in graph J. Additionally, remove signs of all links between w and v_y apart from the sign of $\{w, v_{y_1}\}$. This gives a removal of $l-1$ signs of links. Furthermore, let us remove the signs of all $m-n$ copies of the links that do not appear in the Hamiltonian cycle. This gives a removal of $l+m-n=k$ signs of links in graph J. It is straightforward to verify that after the removal of signs of the above links it holds that $SK'(u, w) = \beta^{n+1}$.

For the other direction, suppose that graph J is a "yes" instance of NSP, i.e., that it is possible to remove the signs of at most k links in the graph, so that upon removal, the graph stays connected (by signed paths) with the value of Signed Katz measure for the target pair of nodes is $\leq r$. Since we can have only the balanced paths between u and w in J, the Signed Katz measure is minimized when this graph is a path graph with u and w as two end nodes. In our particular case, this minimum is exactly β^{n+1}. Given this, let us suppose that after the removal of $m-n+l$ signs of links in graph J we have that $(V_J, E_V \cup H, \sigma(E_V \setminus C))$ is a simple balanced path between u and w. Suppose that in the reaming path, w is connected by a signed edge to v_{y_1} and v_{x_0} is connected to some v_{y_2}. Since u is only connected to v_{x_0} in J, u has to stay connected by a signed link to v_{x_0} after the removal of k signs. From the construction of the graph J it follows that the number of links of J is $m+l+1$. Therefore, after deleting signs of k links the remaining number of links is exactly is $n+1$. Excluding the two links $\{u, v_{x_0}\}$ and $\{w, v_{y_1}\}$, we know that there must be $n-1$ links among the node set X of the original graph G_X. Since the remaining graph is connected, there must exist a Hamiltonian path between x_0 and y_1. Since $\{x_0, y_1\}$ is a link in the graph G_X, we have found a Hamiltonian cycle in G_x, consisting of the Hamiltonian path between x_0 and y_1 together with the link (x_0, y_1), which ends the proof. □

The computational results have been so far negative. We will show now that when the target set of links H is induced by a set of *important* nodes U in the sense that H is simply the set of pairs between all nodes in U, i.e. $H = U \times U$, then some of the problems become tractable.

Let us recall first the definition of *matching* and *maximum matching* of a graph.

Definition 1. Given an undirected graph $G = (V, E)$ a **matching** is a set of pairwise non-adjacent links (none of which are loops), i.e. a set $M \subseteq E$ such that for all distinct $e, e' \in M$ they do not have a common endpoint, i.e. $e \cap e' = \emptyset$. A matching M is a **maximum matching** if it contains the largest possible number of links.

Theorem 4. *Assume that in an instance of* ESCN, $D = E_G$, *and that the target set* H *consists solely of all the links between any two nodes in some set* $U \subseteq V$, *i.e.,* $H = \{\{u, v\} : u, v \in U\}$. *Then, there exists a polynomial-time algorithm for solving* ESCN.

Proof. Let (G, H, k) be a given instance of ESCN. We define $C := \emptyset$ and begin our analysis by considering the graph induced by U, i.e. $G[U]$. We take the maximum matching M_U of this graph and add all the links in the set $E[U] \setminus M_U$ to C.

Moreover, let us consider each node $v \in V_G \setminus U$. We take the set L of all links between v and almost all the elements of U and add L to C as well. Here, by "almost all the elements of U", we mean that we consider all the vertices of U, except one. If there is only a single node in U, then the problem becomes trivial.

Now, depending on the number of elements of C, we output NO, if $|C| > k$. Otherwise, we output C as a positive solution to the problem.

By the fact that M_U is a maximum matching, each node $v \in V \setminus U$ is linked to at most one node in U. Additionally, by the definition of H (recall that we assumed that it is the set of links induced by the set of nodes U), there cannot be any path of two links with undeleted signs in the induced graph $G[U]$. This means that if C is constructed as above, then the number of deleted signs of links is at least $|C|$. From this, it follows that if the algorithm outputs NO, then it is correct. If it outputs YES, then by the fact above, C is a set of at most k links, such that removing their signs results in every pair of vertices in U having no common signed neighbor—similar or dissimilar. Therefore, the positive output of the algorithm denotes a positive instance of the problem, which ends the proof. □

5.2. Parameterized complexity

A natural algorithmic question concerning problems that are NP-hard is to ask whether they are fixed-parameter tractable. Hence, in this section, we analyze the complexity of the problems in question, parameterized by two key parameters of our model:

- the number of links, signs of which may be removed by the attacker, k; and
- the size of the target set H.

We first consider the tractability of ESCN:

Theorem 5. *There exists an algorithm for Eliminating Signed Common neighborhood (*ESCN*) with time complexity of* $\mathcal{O}(c^k + k|E|)$*, where* $c \in (1, 2)$.

Proof. The technique used in this proof is based on the proof of Theorem 1 from Dey and Medya [18]. We adapt it to the setting with signed networks and sign prediction algorithms.

Let $I = (G, H, D, k)$ be an arbitrary instance of ESCN. If for all links $\{u, v\} \in H$ it is the case that $c_s(u, v) \cup c_d(u, v) = \emptyset$, then we output "yes". Otherwise, we may assume that for each signed link $\{u, v\}$ in the target set H and each $x \in V$ such that $\{x\} \subseteq c_s(u, v) \cup c_d(u, v)$ we have that both $\{u, x\}$ and $\{v, x\}$ are in D. Before arguing for the existence of an algorithm, we describe a reduction from ECSN to VC. From I we construct a graph $J = (X, E_X)$ and keep the budget equal to k. For every link $e \in E_V$ we construct a node $x_e \in X$. For all pairs of nodes $x_e, x_f \in X$ we put an link between them, i.e. we put $\{x_e, x_f\} \in E_X$ if there are $u, v, w \in V$ such that $e = \{u, v\}$, $f = \{v, w\}$ and $\{u, w\} \in H$.

Suppose I is a "yes" instance of ESCN. By definition it means there is a set $C \subseteq D$ of links of size k in G such that in the graph $G' = (V, E_G, \sigma(E_G \setminus C))$ for each pair $\{u, w\} \in H$ it holds that $c_s(u, w) \cup c_d(u, w) = \emptyset$. Then the set $Y = \{x_e : e \in C\} \subseteq X$ is a vertex cover of J. Indeed, suppose otherwise. Then there exists en link $\{x_e, x_f\} \in E_X$ such that neither x_e nor x_f is in Y. But then it follows that $x_e = \{u, v\}$, $x_f = \{v, w\}$. But then $c_s(u, w) \cup c_d(u, w) \neq \emptyset$, contrary to the assumption.

For the other direction, suppose J is a "yes" instance of VC. Then, there is a set $Y \subseteq X$ be a vertex cover of J of size k. Then, we define the following set $C \subseteq D$ as the set of links $e \in E$ with $x_e \in Y$. Then for each pair $\{u, w\} \in H$ in the graph $(V, E_V, \sigma(E_V \setminus C))$ we have $c_s(u, w) \cup c_d(u, w) = \emptyset$. Indeed, if it was not the case, then there would be a node v that would be a common neighbor of both u and v. But then it would follow that for $e = \{u, v\}$ and $f = \{v, w\}$, by construction the link $\{x_e, x_f\} \in E_X$ would not be incident to the cover Y, which is impossible. By the fact that VC is FPT w.r.t. the parameter k, [6] the theorem follows. □

Since the proof uses a parameterized reduction from ESCN to VC, by the folklore fact that VC has a P-time approximation algorithm within a factor of 2, we immediately obtain:

Corollary 4. *There is a* P*-time approximation algorithm for optimization of the budget* k *in* ESCN *within a factor of* 2.

Some variants of our problems, however, are hard also in the sense of the parameterized complexity. Recall that a parameterized problem L is para-NP-hard if it is NP-hard even if the parameter k of the problem is fixed.

Theorem 6. ESCN*, when parameterized by the average degree of the graph, is para-*NP*-hard.*

Proof. The technique used in this proof is based on the proof of Theorem 11 from Dey and Medya [18]. We adapt it to the setting with signed networks and sign prediction algorithms.

The proof goes by an easy reduction from ESCN. For a given instance of ESCN the reduction produces an additional instance of ESCN with a constant average degree. To see this, let (G, H, D, k) be an instance of ESCN. Assume that the number of nodes of G is n, i.e., $|V_B| = n$. Additionally, pick any node $v \in V_G$, and define the following instance (G^*, H^*, D^*, k^*) of ESCN:

$$V_{G^*} = V_G \cup \{v_i : i \in \{1, \ldots, n^2\}\},$$
$$E_{G^*} = E_G \cup \{v, v_1\} \cup \{()\},$$

and

$$\sigma^* = \sigma \cup \{\langle (v, v_1), + \rangle\} \cup \{\langle (v_i, v_{i+1}), + \rangle : i \in \{1, \dots, n-1\}\}.$$

Additionally, let $D^* = D$, $H^* = H$, and finally, keep $k^* = k$.

The main observation is that the average degree of the newly constructed signed graph G^* is bounded by 2, namely:

$$avg_{G^*}(d) \leq \frac{2n^2}{n^2 + n} \leq 2.$$

It is now easy to notice that the two instances are equivalent – implication in one direction is immediate, and for the other, it is sufficient to observe that if $C^* \subseteq E^*$ is a solution for (G^*, H^*, D^*, k^*), then its restriction:

$$C := C^* \cap E$$

is a solution for (G, H, D, k) as well. This ends the proof. □

Theorem 7. NTSP *for local signed similarity measures, when parameterized by* k*, is* W[1]*-hard.*

Proof. The technique used in this proof is based on the proof of Theorem 8 from Dey and Medya [18]. We adapt it to the setting with signed networks and sign prediction algorithms.

The proof goes by a parameterized reduction from PARTIAL VERTEX COVER (PVC), where given a graph $G = (X, E_X)$ and two integers $k, p \in \mathbb{N}$, the problem is to decide if there is a set U of at most k nodes such that there is a set P of at least p links such that each link from P is incident to at least one node from U. It is well-known that PVC is W[1]-hard, when parameterized by k [15].

Let $I = (G, k, p)$ be an instance of PVC. Assume $|X| = n$ and fix any numbering of X, that is let $X = \{x_1, \dots, x_n\}$, and suppose $|E_X| = m$.

First, we construct a signed graph $J := (V, E_V \cup H, \sigma(E_V))$. The set of nodes V consists of the following: for each node $x_i \in X$ construct its copy $v_i \in V$, and additionally define a fresh single element $w \in V$. The set of links E_V and their signs σ is defined as follows: for each link $\{x_i, x_j\} \in E_X$ we construct two positive links $\{w, v_i\}$ and $\{w, v_j\}$. Let the target set H be again the copy of the set E_X, put $D := E_V$. Finally, let the budget of links in the constructed instance of NTSP be equal to k, and let the sum r of the similarity scores of links from H be equal to $m - p$.

To see the reduction is correct, first assume I is a "yes" instance of PVC. Let $U \subseteq X$ be k-sized set $\{u_1, \dots, u_k\}$ that covers a p-sized set P of links in G, namely for each link $e_j = \{x, y\}$ in P, either x is equal to some $u_i \in U$, or y is. Then we get that for each link $\{x, y\} \in P$ it holds that after the removal of the signs of the links $\{w, u_i\}$ for $u_i \in U$, the updated signed common neighborhood of x and y, is equal to 0, so $\mathrm{SJ}'(x, y) = SPA'(x, y) = SCN'(x, y) = 0$. But then, since there are p links $\{x, y\} \in E_G$ covered by U, the sum of the sign similarity scores of links in H is no greater than $m - p$, as required.

Now, suppose J is a "yes" instance of NTSP. Then, there is a set of links $D = \{\{w, v_i\} : 1 \leq i \leq k\}$ of size k such that after removing the signs of the links from D, the sum $\sum_{\{u,v\} \in H} sim'(u, v) \leq m - p$. But then if we consider the set of nodes $U = \{v_i : 1 \leq i \leq k\} \subseteq X$, then by definition it must be the case that each link $\{x, y\} \in E_X$ such that $\{w, x\} \in D$ or $\{w, y\} \in D$ is incident to some node from D. But by construction, there are at least p links $\{x, y\} \in E_X$ satisfying this condition, so the set U of nodes covers at least p-sized set of links in G, which makes I a "yes"instance of PVC, and ends the proof. □

Finally, it turns out that the general form of NSCN (e.g. without further restriction on r) is the hardest of the problems analyzed. We namely have that:

Theorem 8. NSCN *parametrized by* k *is* W[2]*-hard.*

Proof. The technique used in this proof is based on the proof of Theorem 9 from Dey and Medya [18]. We adapt it to the setting with signed networks and sign prediction algorithms.

The proof goes by a parameterized reduction from UNIFORM SET COVER (USC), where given an integer $k \in \mathbb{N}$, a set X, and family $\mathcal{F} \subseteq \mathcal{P}(X)$ with the property that the size of the subfamily $\mathcal{F}_x = \{A \in \mathcal{F} : x \in A\}$ is the same for each $x \in X$, the problem is to decide if there exists a subfamily $\mathcal{F}^* \subseteq \mathcal{F}$ of size at most k such that $\bigcup_{A \in \mathcal{F}^*} A = X$. An instance of USC will be denoted as $I = (X, \mathcal{F}, k)$. It is known that USC parameterized by k is W[2]-hard. For an FPT-reduction, let I be an instance of USC, and let $c = |\mathcal{F}_x|$ be the uniform size of subfamilies containing x for each $x \in X$. We construct the following instance $J := (V, E_V \cup H, \sigma(E_V), D, k^*, r)$ of NSCN. Let the set of nodes consist of: a single root node w, a set of copies v_x of the elements $x \in X$, and a set of copies u_A for all $A \in \mathcal{F}$. For the set of links of J take the sum of $\{\{w, u_A\} : A \in \mathcal{F}\}$ and $\{\{v_x, u_A\} : x \in A\}$, and assume the signs of all the links are positive. Further, define the target set of links H as the set $\{\{w, v_x\} : x \in X\}$. Finally, put $r := c - 1$, $k^* = k$, and let $D = E_V$.

For the proof that the reduction is correct, first observe that by construction, for each link $\{w, v_x\} \in H$, the signed common neighborhood $\mathrm{SCN}(w, v_x) = c$. Now suppose that I is a "yes" instance of USC. This means there is a subfamily $\mathcal{F}^* \subseteq \mathcal{F}$ with $|\mathcal{F}^*| \leq k$, that covers X, i.e. for each $x \in X$ there is an $A \in \mathcal{F}^*$ such that $x \in A$. Define the following set of links, signs of which shall be removed from J: $C := \{\{w, u_A\} : A \in \mathcal{F}^*\}$. Then, since for each $x \in X$, there is an $A \in \mathcal{F}^*$ such that $\{v_x, u_A\} \in E_V$, it holds that for each pair $\{w, v_x\} \in H$ we have $\mathrm{SCN}(w, v_x) \leq c - 1$, since w is a common neighbor of v_x and u_A for $A \ni x$, and at least one link

between w and such u_A must have had its sign removed by the covering property of $\mathcal{F}^*$. But this means exactly that J is a "yes" instance of NSCN.

Now assume J is a "yes" instance of NSCN. Let $C \subseteq E_V$ be a k-sized solution. In I, define $\mathcal{F}^* := \{A \in \mathcal{F} : \{w, u_A\} \in C\} \cup \{A \in \mathcal{F} : \exists x \in X \ \{v_x, u_A\} \in C\}$. By definition, the size of $\mathcal{F}^*$ is at most k. By the fact that C is a solution of NSCN in J, for each target link $\{w, v_x\}$, a sign of at least one link of the form $\{w, u_A\}$ with $x \in A$ or of the form $\{v_x, u_A\}$ must have been removed, so that $\mathrm{SCN}'(w, v_x) \leq c - 1$. From this it directly follows that for every $x \in X$ there is at least one $A \in \mathcal{F}^*$ with $x \in A$, namely these are all A such that u_A is a common neighbor of v_x and w in J. Hence, $\mathcal{F}^*$ is a cover of X, and I is a "yes" instance of USC. □

Since NSCN is a sub-problem of NSP for local metrics, it trivially follows that the unrestricted version of NSP for local signed similarity measures is W[2]-hard as well:

Corollary 5. *The problem* NSP *is* W[2]*-hard with respect to the parameter* k.

We leave parameterized complexity of RSP w.r.t. k as an open problem.

For the next result, we need to recall a seminal result by Lenstra:

Theorem 9 *(Lenstra's Theorem [15], p. 130, and Lenstra [30]). Integer Linear Programing is fixed-parameter tractable when parameterized by the dimension of the space, i.e., by the number of variables of the problem.*

Since many NP-hard problems can be expressed in terms of integer linear programs, Lenstra's Theorem provides quite a general tool for designing fixed-parameter algorithms. We use this tool directly below, to prove the following:

Theorem 10. NTSP*, when parameterized by* $|H|$*, is fixed-parameter tractable.*

Proof. The technique used in this proof is based on the proof of Theorem 3 from Dey and Medya [18]. We adapt it to the setting with signed networks and sign prediction algorithms.

Let (G, H, D, k, r) be an instance of the optimization version of NTSP. Without loss of generality we may assume that for each pair $\{u, v\} \in H$ and each node $w \in V_G$, if $\{u, w\}, \{v, w\} \in E_G$, then $\{u, w\}, \{v, w\} \in D$.

We can partition the set of nodes V_G by an equivalence relation $\approx$ defined as follows.

First, by $\varphi(u, v, w, w')$ denote the formula expressing a fact that u and v are both common neighbors of w and w', i.e. a formula:

$$\{u, w\} \in E \ \& \ \{u, w'\} \in E \ \& \ \{v, w\} \in E \ \& \ \{v, w'\} \in E.$$

Let $\overline{\varphi}(u, v, w, w')$ denote the formula expressing the fact that neither u nor v is a common neighbor of w and w', i.e. a formula:

$$\Big(\{u, w\} \notin E \ \vee \ \{u, w'\} \notin E\Big) \ \& \ \Big(\{v, w\} \notin E \ \vee \ \{v, w'\} \notin E\Big).$$

Further, by $\psi(u, v, w, w')$ denote the formula expressing a fact that u and v are both common neighbors of w and w' and they both make the absolute value of SCN of w and w' increase, i.e. they either both are in $c_s(w, w')$ or both are in $c_d(w, w')$ which means that the signs of $\{u, w\}$ and $\{u, w'\}$ are identical, exactly when the signs of $\{v, w\}$ and $\{v, w'\}$ are. In other words, $\psi(u, v, w, w')$ is:

$$\varphi(u, v, w, w') \ \& \ \big(\sigma(\{u, w\}) = \sigma(\{u, w'\}) \text{ iff } \sigma(\{v, w\}) = \sigma(\{v, w'\})\big).$$

Additionally, by $\gamma(u, v, w, w')$ denote a formula expressing a fact that u and v are both common neighbors of w and w' and they do not change the SCN of w and w', i.e. exactly one of u and v is in $c_s(w, w')$ whereas the second one is in $c_d(w, w')$ which means that the signs of $\{u, w\}$ and $\{u, w'\}$ are identical, and the signs of $\{v, w\}$ and $\{v, w'\}$ are not, or vice versa. In other words, $\gamma(u, v, w, w')$ is:

$$\varphi(u, v, w, w') \ \& \ \big(\sigma(\{u, w\}) = \sigma(\{u, w'\}) \text{ iff } \sigma(\{v, w\}) \neq \sigma(\{v, w'\})\big).$$

Now, for any $u, v \in V_G$: $u \approx v$ if and only if for each link $\{w, w'\} \in H$ it holds that:

$$\psi(u, v, w, w') \ \vee \ \big(\gamma(u, v, w, w') \ \vee \ \overline{\varphi}(u, v, w, w')\big).$$

To illustrate the above definition, let the size of H be equal to m, and put $H = \{e_1, \ldots, e_m\}$. Then, for each node $v \in V_G$ define the sign of v with respect to $e_i = (w_i, w_i')$ as:

$$v_i = \begin{cases} 1 & \text{if } (\{u, w_i\} \in E \ \& \ \{u, w_i'\} \in E) \ \& \ \sigma(\{u, w_i\}) = \sigma(\{u, w_i'\}) \\ 0 & \text{if } (\{u, w_i\} \notin E \ \vee \ \{u, w_i'\} \notin E) \\ -1 & \text{if } (\{u, w_i\} \in E \ \& \ \{u, w_i'\} \in E) \ \& \ \sigma(\{u, w_i\}) \neq \sigma(\{u, w_i'\}). \end{cases}$$

Now, we can identify each node in the graph with the sequence $(v_1, \ldots, v_m)$. Then we can see that nodes u and v are $\approx$-equivalent if for each $i \leq m$ we have that:

$|u_i + v_i| \neq 1$.

Therefore, it follows that there are exactly 2^m equivalence classes on the spaces of all possible sequences $(u_i)_{1 \leq i \leq m}$, since each class can be identified with a binary sequence of length $m = |H|$ over $\{0, 2\}$, as these are the possible values of $|u_i + v_i|$. Hence, each class can be further identified with a subset of H. For each possible class α consider the number $n_G(\alpha)$ which is the number of nodes of G in the class α. It is easy to see that each node u from a given class α and a given subset $K \subseteq H$ for each $\{w, w'\} \in K$ such that u is a common neighbor of w and w' one of the following four possibilities happens with respect to the optimal solution of the problem:

(a) both $\{u, w\}$ and $\{u, w'\}$ are elements of the optimal solution,
(b) $\{u, w\}$ is and $\{u, w'\}$ is not an element of the optimal solution,
(c) $\{u, w'\}$ is and $\{u, w\}$ is not an element of the optimal solution,
(d) neither $\{u, w\}$ nor $\{u, w'\}$ are elements of the optimal solution.

For each $K \subseteq H$, let $P_\alpha(K) := \{\text{a,b,c,d}\}^K$ denote the set of all sequences of possibilities of relating each node in the class α to the optimal solution with respect to the links in K, and let $P^*_\alpha(K) \subseteq P_\alpha(K)$ be its subset containing the sequences that are not constantly equal to d.

Observe that by the definition of α, we can forget about the set K, and treat α abstractly.

Then, it is now easy to formulate the RTSP as an ILP problem.

For each class α and every set $P_\alpha(K)$, let $t(\alpha)$ be the number of links, signs of which are deleted in $P_\alpha(K)$ and for any $K \subseteq H$ (or any $\alpha \in \mathcal{A}$), for any $P \in P_\alpha(K)$ let $s(\alpha, P)$ be the number of nodes from the class α that realize the same sequence of possibilities of relating each node in the class α to the optimal solution with respect to the links in K. Consider the following ILP:

$$\sum_{\{u,v\} \in H} |\text{SCN}(u, v)| \leq r, \tag{3}$$

$$\sum_{\alpha \in \mathcal{A}, P \in P_\alpha(K)} t(\alpha) s(\alpha, P) \leq k, \tag{4}$$

$$\forall K \subseteq H \quad \sum_{P \in P_\alpha(K)} s(\alpha, P) = n_G(\alpha), \tag{5}$$

$$\forall \{u, v\} \in H \quad |\text{SCN}(u, v)| = \sum_{\alpha \in \mathcal{A}} n_G(\alpha) - \sum_{P \in P^*_\alpha(K)} s(\alpha, P). \tag{6}$$

By Lenstra's Theorem, i.e., Theorem 9, it is thus FPT. □

The ILP formulation of the proof above immediately gives the following corollaries:

Corollary 6. ESCN*, when parametrized by $|H|$ is fixed-parameter tractable.*

Corollary 7. NSCN*, when parameterized by $|H|$, is fixed-parameter tractable.*

Corollary 8. *Let sim be a local similarity measure such that, for any signed graph $G = (V, E, \sigma)$ and any $u, v \in V$, it holds that:*

$$|c_s(u, v)| - |c_d(u, v)| = 0 \implies sim(u, v) = 0 \tag{7}$$

Then, if $r = 0$, NSP *is* FPT *with respect to the parameter $|H|$.*

Our results on the fixed-parameter tractability presented in this section provide feasible algorithms for both parameterized variants of the NSP and ESCN problems that we consider, the NSCN problem parameterized by the size of the target set, and the NSCN problem parameterized by the size of the budget, under the assumption that the threshold is 0. In the next section, we propose heuristics for the ESCN and RSP problems.

6. Heuristics

In this section, we first propose three heuristics for dealing with the ESCN, NTSP, and RSP problems, and we evaluate their effectiveness empirically on both synthetic and real-life datasets.

6.1. CTSR heuristic

A heuristic for eliminating the signed common neighborhood is to serve an attacker wishing to hide the signs of links $\{u, w\} \in H$ by removing a sign from a closed triangle of u, v, w, hence, it is called Closed Triangles Sign Removal (CTSR). Specifically, given $\{u, w\} \in H$, it deletes the sign of each link $\{u, v\}$ such that there exists a node $v \in V$ and that $\{v, w\}$ is also a signed link in G.

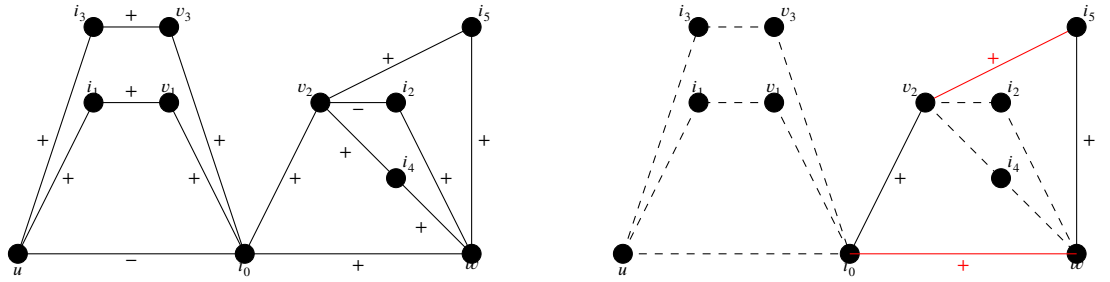


Fig. 9. An instance of ESCN before (left) and after (right) the CTSR heuristic. Let the target set be composed of the pairs: $\{u, w\}$, $\{u, v_1\}$, $\{u, v_3\}$, and $\{w, v_2\}$. First, the preprocessing computes the set $B(H)$ of links, the signs of which balance each other. We highlight the links in $B(H)$ using dashed lines in the preprocessed graph. Next, the preprocessing removes from the graph the signs of the links in $B(H)$. After this preprocessing, the CTSR heuristic removes the signs of links for which the *gain* that we can achieve by doing so is maximal. In the preprocessed graph on the right we highlight two such links in red.

First, we preprocess the graph to make the heuristics more effective. For each $\{u, w\} \in H$ with $\mathrm{CN}(u, w) > |\mathrm{SCN}(u, w)|$, if $\mathrm{CN}(u, w) - |\mathrm{SCN}(u, w)| = 2l > 0$ (observe that this difference has to be an even number), then there exist exactly $2l$ pairs of links e_i, f_i, $i \leq n$ such that:

$$
\begin{aligned}
&e_1 = \{u, v_1\},\ f_1 = \{w, v_1\},\\
&e_2 = \{u, v_2\},\ f_2 = \{w, v_2\},\\
&\dots\\
&e_{2l} = \{u, v_{2l}\},\ f_{2l} = \{w, v_{2l}\},
\end{aligned}
$$

and such that for each $i \leq l$ we have $\sigma(e_i) = \sigma(f_i)$, and for each $i \in [l+1, 2l]$ $\sigma(e_i) \neq \sigma(f_i)$. For each pair $\{u, w\} \in H$, call the set of these links $B(\{u, w\})$ (B as in balance, since the links in B balance each other for u and w). We transform the graph $G = (V, E \cup H, \sigma)$ into the graph $G' = (V, E \cup H, \sigma(E \setminus B(H))$, where $B(H) = \sum_{\{u,w\} \in H} B(\{u, w\})$. Then we apply the following algorithm to the reduced graph G'. The algorithm will run until the budget of k links to delete is exhausted. In principle, it is possible that some of the links from $B(H)$ also have signs that need to be removed in order to fully neutralize the signed common neighborhood. But if this is the case, and t such links need to have their signs removed, then it simply means that, if the algorithm manages to neutralize the signed common neighborhood in at most $k - t$ steps, then it can perform this task in k steps as well.

The pseudocode for the heuristic is presented in Algorithm 1. The input is a preprocessed instance of ESCN, i.e., (G', H, D, k). In the algorithm, we use function g which measures the *gain* that we can achieve by removing the sign of a particular link. The idea here is that the procedure deletes the signs of links from closed triangles of nodes in G'. The more triangles are removed with a single removal of a sign of a given link, the better, and the function g is there precisely to inform us of the possibility to get rid of more than one triangle via the removal of a single sign. Fig. 9 presents an example of applying the CTSR heuristic.

Algorithm 1 CTSR Heuristic.

Input: A preprocessed instance of ESCN (G', H, D, k).

$D' := \{\{v, w\} \in D : (\exists u \in N(w)\ \{u, v\} \in H) \vee (\exists u \in N(v)\ \{u, w\} \in H)\}$,
for $i = 1, \dots, k$ **do**
  for $\{v, w\} \in D'$ **do**
    $g(\{v, w\}) := 0$;
  end for
  for $\{u, w\} \in H$ **do**
    for $v \in CN(u, w)$ **do**
      if $\{v, w\} \in E'$ & $\{v, u\} \in E'$ **then**
        if $\{v, w\} \in D'$ **then**
          $g(\{v, w\}) := g(\{v, w\}) + 1$;
        end if
        if $\{v, u\} \in D'$ **then**
          $g(\{v, u\}) := g(\{v, u\}) + 1$;
        end if
      end if
    end for
  end for
  $\{v^*, w^*\} := \arg\max_{\{v,w\} \in D'} g(\{v, w\})$
  if $g(\{v^*, w^*\}) > 0$ **then**
    $\sigma := \sigma \setminus \{\sigma(\{v^*, w^*\})\}$
  end if
end for

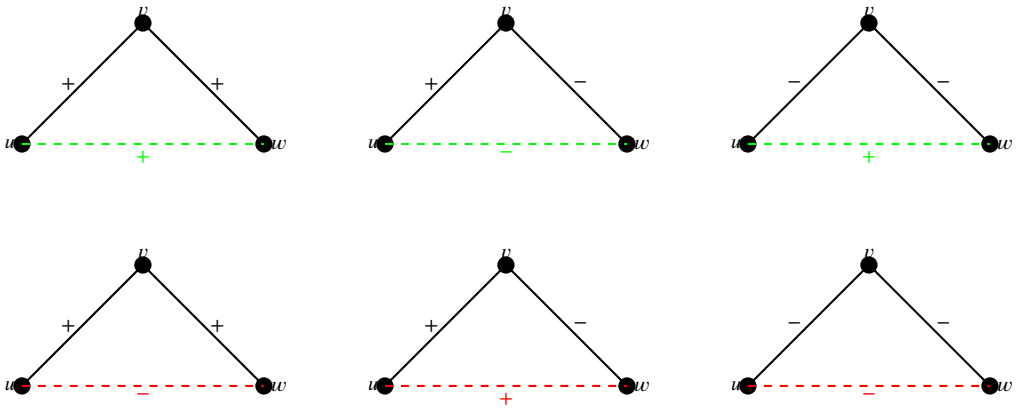


Fig. 10. Three types of balanced (top) and unbalanced triangles (bottom). Suppose $\{u, w\}$ is an element of the target set of links H. In the BTSR heuristic, in order to reverse the SCN score of the pair $\{u, w\}$, the algorithm removes the sign of $\{u, v\}$ or $\{w, v\}$ in the balanced triangles, and avoid removing any signs of links in the case of unbalanced triangles.

6.2. BTSR heuristic

Here we consider a RSP (reversing sign prediction) variant of the problem of attacking similarity-based sign prediction in which the goal of the attacker is not to neutralize the similarity score between nodes u, w with $\{u, w\} \in H$, but to reverse the sign of the value of similarity score of $\{u, w\}$ in the full network.

Our heuristic is motivated by the aforementioned balance theory which indicates—in graph-theoretic language—that in signed networks the triangles of nodes connected by links should exemplify the principles that *the friend of my friend is my friend*, *the friend of my enemy is my enemy*, or *the enemy of my friend is my enemy*, and *the enemy of my enemy is my friend*. Under a natural interpretation of these principles, a triangle in a signed undirected graph is balanced (and as such, presumably more plausible to encounter in a real network) if there are one or three positive links in it. This can be further interpreted in the context of sign prediction as the principle that, if we have a triangle of nodes $\{u, v, w\}$ and we see that $\sigma(\{u, v\}) = \sigma(\{v, w\})$, then it is likely that the link $\sigma(\{u, w\}) = +$, and that if $\sigma(\{u, v\}) \neq \sigma(\{v, w\})$, then it is likely that $\sigma(\{u, w\}) = -$.

The Balanced Theory Sign Removal (BTSR) heuristic can serve the attacker wishing to hide the sign of the link $\{u, w\} \in H$ as follows. Remove the sign of $\{u, v\}$ if there exists a node $v \in V$ with $\{v, u\} \in E$ and $\{v, w\} \in E$ (thus forming a closed triangle) that satisfies one of the following conditions imposed on the signs of the links:

- either $\sigma(\{u, v\}) = \sigma(\{v, w\})$ and $\sigma(\{u, w\}) = +$,
- or $\sigma(\{u, v\}) \neq \sigma(\{v, w\})$ and $\sigma(\{u, w\}) = -$.

The pseudocode of the BTSR heuristic is presented in Algorithm 2, which can be found in the appendix. Fig. 10 further explains the intuition behind the heuristic.

6.3. Tally heuristic

We now present an alternative heuristic for both the NSP and the RSP problems, namely the Tally heuristic. While the CTSR and the BTSR heuristics are focused on the positive contribution of the removal of a sign from the network's link (i.e., how said removal brings the SCN score of an incident link from H closer to the desired value), the Tally heuristic takes into consideration the negative contributions as well. Notice how if a given link from D is incident with multiple links from H, the removal of its sign can bring the score of some links from H closer to the desired value, while at the same time doing the opposite for some other links in H. The Tally heuristic counts all such changes and removes the sign of the link with the best balance.

One of the arguments of the Tally heuristic is the positive contribution condition $\phi(u, w, v)$ that decides whether the removal of a sign from link $\{u, v\}$ or $\{w, v\}$ has the desired effect on the link $\{u, w\} \in H$. We use the following formulas for ϕ:

- for the NSP problem

$$\phi(u, w, v) = \big(\mathrm{SCN}(u, w) > 0 \wedge v \in c_s(u, v)\big) \vee \big(\mathrm{SCN}(u, w) < 0 \wedge v \in c_d(u, v)\big)$$

- for the RSP problem

$$\phi(u, w, v) = \big(\sigma(u, w) = + \wedge v \in c_s(u, v)\big) \vee \big(\sigma(u, w) = - \wedge v \in c_d(u, v)\big).$$

Notice how, for the NSP problem, the removal of a sign from link $\{u, v\}$ or $\{w, v\}$ is deemed to have a desirable effect if the SCN score of the link $\{u, w\} \in H$ is non-zero, and the removal brings it closer to zero. Otherwise, i.e., if the removal brings the SCN score

Algorithm 2 BTSR Heuristic.

Input: (G, H, D, k), where G is a signed network, H is the target set of pairs to be hidden, $D \subseteq E$ is the unpreprocessed link set, the signs of which can be removed, and k is the number of signs of links that can be removed.

$D' := \{\{v,w\} \in D : (\exists u \in N(w)\ \{u,v\} \in H\ \&\ (\sigma(\{u,w\}) = \sigma(\{v,w\})\ \&\ \sigma(\{u,v\}) = +)\ (\vee\ \sigma(\{u,w\}) \neq \sigma(\{v,w\})\ \&\ \sigma(\{u,v\}) = -))\ \vee\ (\exists u \in N(v)\ \{u,w\} \in H\ \&\ (\sigma(\{u,v\}) = \sigma(\{v,w\})\ \&\ \sigma(\{u,w\}) = +) \vee (\sigma(\{u,v\}) \neq \sigma(\{v,w\})\ \&\ \sigma(\{u,w\}) = -))\}$,
for $i = 1, \ldots, k$ **do**
  for $\{v,w\} \in D'$ **do**
    $g(\{v,w\}) := 0$;
  end for
  for $\{u,w\} \in H$ **do**
    for $v \in CN(u,w)$ **do**
      if $\{v,w\} \in E'$ & $\{v,u\} \in E'$ **then**
        if $\{v,w\} \in D'$ **then**
          $g(\{v,w\}) := g(\{v,w\}) + 1$;
        end if
        if $\{v,u\} \in D'$ **then**
          $g(\{v,u\}) := g(\{v,u\}) + 1$;
        end if
      end if
    end for
  end for
  $\{v^*, w^*\} := \arg\max_{\{v,w\} \in D'} g(\{v,w\})$
  if $g(\{v^*, w^*\}) > 0$ **then**
    $\sigma := \sigma \setminus \{\sigma(\{v^*, w^*\})\}$
  end if
end for

further away from zero, it is deemed to have an undesirable effect. Similarly, for the RSP problem the removal of a sign is deemed to have a desirable effect if it decreases the SCN score of a link in H with the plus sign, or it increases the SCN score of a link in H with the minus sign. Otherwise, the removal is deemed to have an undesirable effect.

The pseudocode of the Tally heuristic is presented in Algorithm 3.

Algorithm 3 Tally heuristic.

Input: a signed network G, the target set of pairs to be hidden H, the set of links the signs of which can be removed $D \subseteq E$, the number of signs that can be removed k, and the positive contribution condition $\phi(u,w,v) : V \times V \times V \to \{0,1\}$.

$D' := \{\{v,w\} \in D : (\exists u \in N(w)\ \{u,v\} \in H) \vee (\exists u \in N(v)\ \{u,w\} \in H)\}$
for $i = 1, \ldots, k$ **do**
  for $\{v,w\} \in E$ **do**
    $g(\{v,w\}) := 0$
  end for
  for $\{u,w\} \in H$ **do**
    for $v \in c_s(u,v) \cup c_d(u,v)$ **do**
      if $\phi(u,w,v)$ **then**
        $g(\{v,u\}) := g(\{v,u\}) + 1$
        $g(\{v,w\}) := g(\{v,w\}) + 1$
      else
        $g(\{v,u\}) := g(\{v,u\}) - 1$
        $g(\{v,w\}) := g(\{v,w\}) - 1$
      end if
    end for
  end for
  $\{v^*, w^*\} := \arg\max_{\{v,w\} \in D'} g(\{v,w\})$
  if $g(\{v^*, w^*\}) > 0$ **then**
    $\sigma := \sigma \setminus \{\sigma(\{v^*, w^*\})\}$
  end if
end for

7. Experimental evaluation

In this section, we evaluate the effectiveness of the heuristics presented in the previous section using simulations on both randomly-generated and real-life networks.

7.1. Datasets

In our simulations we consider the following real-life signed networks datasets:

- **Bitcoin Alpha** [28]—the network of trust and distrust relations between the users of the Bitcoin Alpha trading platform,

Table 3
Information about real-life datasets used in the simulations.

Dataset	Nodes	Links	Negative signs percentage
Bitcoin Alpha	3,775	14,120	9.59%
Bitcoin OTC	5,875	21,489	14.94%
Wikipedia RFA	11,379	181,041	26.75%
Slashdot	79,116	467,731	25.38%
Epinions	119,130	704,267	17.10%
Wikipedia trust	137,740	715,334	12.25%

- **Bitcoin OTC** [28]—the network of trust and distrust relations between the users of the Bitcoin OTC trading platform,
- **Wikipedia RFA** [52]—the network of positive and negative votes in the Wikipedia request for adminship process,
- **Slashdot** [29]—the network of friend-or-foe relations between the users of a technology news site Slashdot,
- **Epinions** [32]—this network of who-trust-whom relations between the users of a general consumer review site Epinions,
- **Wikipedia trust** [35]—the network of interactions between the users of the English Wikipedia who edited pages about politics.

We preprocess each real-life dataset in the following way.

1. In this work, we deal with undirected networks, hence if a given dataset is directed, we convert it to an undirected network. If for a given pair of nodes there exists a link in only one direction, we replace it with an undirected link with the same weight. Alternatively, if for a given pair of nodes there exist links in both directions, we replace them with an undirected link with the weight being the average of the two weights.
2. We set the sign of each link to plus if it has a positive weight, and to minus if it has a negative weight.
3. Some of the datasets are not connected, with the giant component containing over 90% of the nodes. In these cases, we perform the computation only on the giant component.

Table 3 provides information about the preprocessed datasets.
In our simulations, we also consider the following models of randomly-generated networks:

- **Barabási-Albert** networks [2]—generated using the preferential attachment model,
- **Erdős-Rényi** networks [19]—where a link is created between each pair of nodes with a given probability,
- **Watts-Strogatz** networks [50]—meant to represent a small-world structure with a short average distance between any pair of nodes. We set the rewire probability to $\frac{1}{4}$.

Unless stated otherwise, we consider randomly generated networks with 2000 nodes and the average degree of 30. We set the signs of 10% of the links (chosen uniformly at random) to minus, and set the signs of the remaining 90% to plus. We chose this value as it reflects the percentage of negative links in the real-life signed networks datasets.

7.2. Experimental procedure

We now describe the experimental procedure of our simulations. Given an undirected signed network, we first select the set of links H the sign of which we will attempt to hide. We consider three different criteria of selecting the set H:

- **high score**—we select the links with the greatest values of the SCN score,
- **local**—we first randomly select a node with degree at least 10, and then we select the links uniformly at random out of those that are incident with the neighbors of the node (as a result, we obtain links from a local neighborhood of the node),
- **any score**—we select the links uniformly at random out of all the links in the network.

Unless stated otherwise, in the simulations we select 50 links to be the elements of H.

We then use the heuristics presented in the previous section in an attempt to hide the signs of the links in H. Unless stated otherwise, we set the hiding budget to be the same as the size of H. During the hiding process, we record the sum of the absolute values of the SCN scores for the NSP problem instances, and the sum of differences between the initial SCN scores and the current SCN scores for the RSP problem instances. Observe that because we evaluate the heuristics using sums of (the absolute values of) the SCN scores, the heuristics we provide can also be used for solving the NTSP problem as well.

For the real-life network datasets, we run the process for 1000 different sets H per selection criterion. For each random network generation model, we generate 100 different networks, and for each of them we select 10 different sets H per selection criterion.

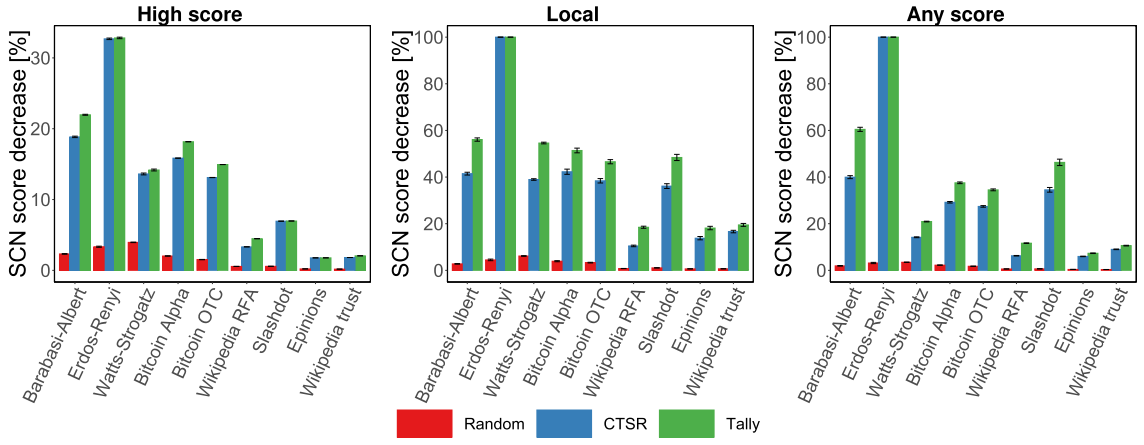


Fig. 11. The decrease in the sum of the absolute SCN scores of the links in H after the entire hiding process for the NSP problem instances (larger values indicate better effectiveness of the heuristic). The left plot corresponds to H selected as the links with the greatest SCN scores, the middle plot to H selected as the links incident with the neighbors of a single node, while the right plot to links H selected uniformly at random. Error bars represent the 95%-confidence intervals.

7.3. Basic simulation results

Here, we compare the performance of our heuristics in real-life and randomly-generated networks. First, let us focus on the NSP problem instances. Fig. 11 compares the drop in the SCN scores at the end of the hiding process, while Fig. 12 presents the effectiveness of the heuristics throughout the hiding process. As can be seen, the CTSR and Tally heuristics are both significantly more effective than the random approach, with the Tally heuristic exhibiting slightly better performance than CTSR. Moreover, hiding links that are chosen uniformly at random, or links in the local vicinity of a given node, is on average much more effective than the attempt to hide links with very high SCN scores from the beginning.

Next, we consider the RSP problem instances. Fig. 13 compares the difference in the SCN values at the end of the hiding process, while Fig. 14 compares the effectiveness of the heuristics throughout the hiding process. Similarly, as in the results for the NSP setting, both heuristics presented in the previous section are significantly more effective than the random alternative, with the Tally heuristic being the most successful of all. The performance is best for links selected due to their high SCN scores, although it is worth noting that these are exactly the edges that need to have their scores most significantly decreased in order to become hidden.

7.4. Varying the parameters of the problem instances

In the experiments presented thus far, we considered randomly-generated networks with 2000 nodes and an average degree of 30. Moreover, the size of H was always 50. We now present results for networks with varying numbers of nodes and average degrees, as well as for H with varying sizes.

As can be seen in Fig. 15, hiding signs in the NSP setting is generally more effective in sparse networks. As for the size of the network, trends observed in our simulations are less uniform. In Watts-Strogatz networks, for example, the hiding process is more effective in smaller networks, while in their Erdős-Rényi counterparts the opposite is true. Interestingly, in Barabási-Albert networks, hiding is more effective in larger networks when the links to be hidden are selected uniformly at random or from the local vicinity of a node, but more effective in smaller networks when said links are those with the greatest SCN scores. As for the size of the H set, in most cases it is easier to hide the signs of edges from a larger set, with the only exception being edges selected uniformly at random from the Barabási-Albert networks.

Fig. 16 presents the results of the RSP simulations. The general trends remain consistent with those observed for the NSP setting: in the vast majority of cases, the hiding process is more effective in smaller and denser networks, and for larger sets H.

7.5. Comparing to optimal strategies

In the experiments presented thus far, we used networks too large and too dense to compute optimal hiding strategies for them. We now present results for networks with 500 nodes and an average degree of 5, where we intend to hide the sign of 5 edges at a time. In these networks, we are able to compute the optimal solution by evaluating all possible subsets of edges that are allowed to be removed by our heuristics.

Fig. 17 presents the results for the NSP problem. As can be seen, when hiding edges with the greatest scores, both the CTSR and the Tally heuristics achieve performances relatively close to that of the optimal strategy, i.e., between 80% and 100% of the opti-

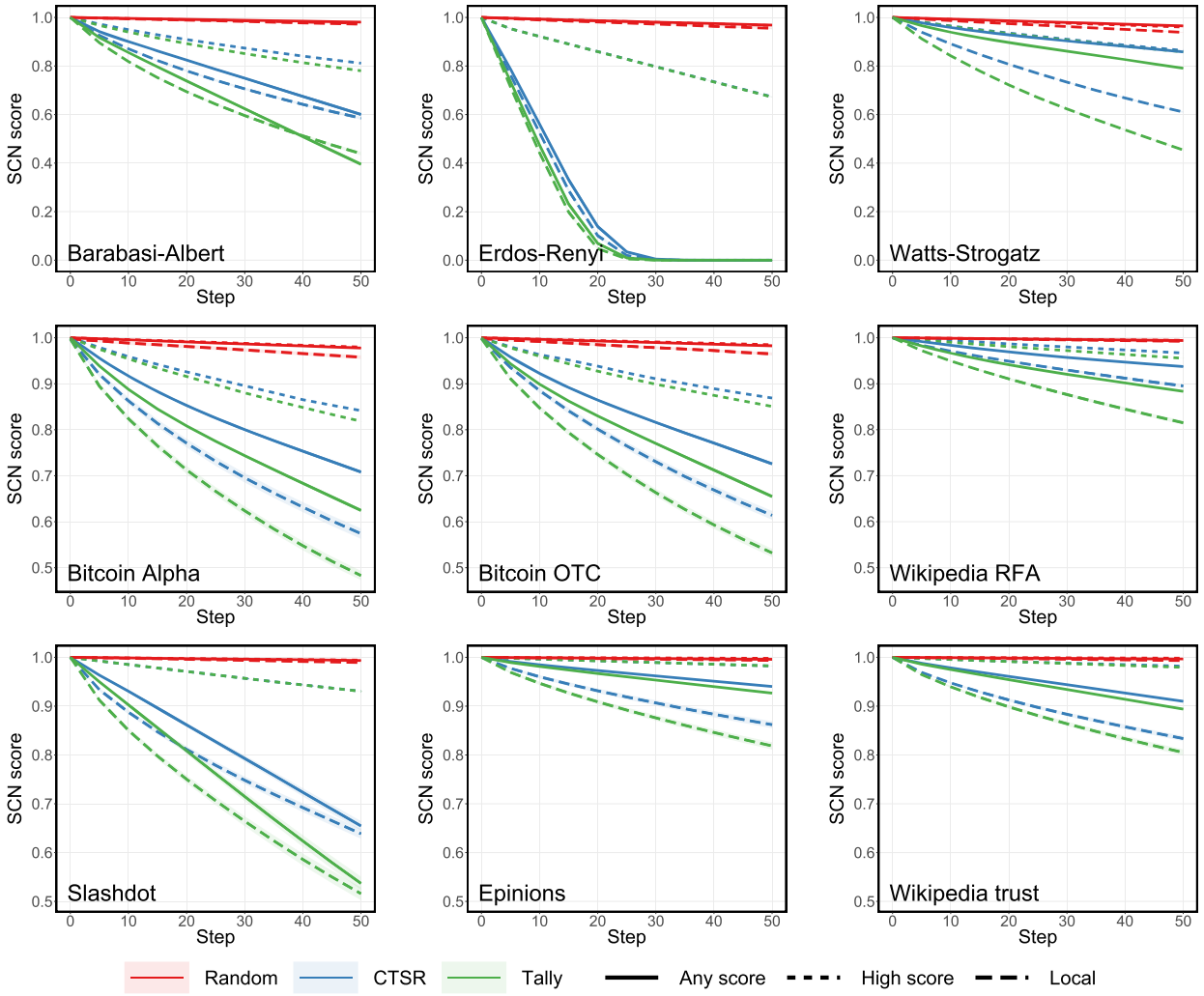


Fig. 12. A comparison of the heuristics' effectiveness for the NSP problem instances throughout the hiding process. In each plot, the x-axis represents the number of executions of the heuristic, while the y-axis represents the relative sum of the absolute SCN scores of the links in H (smaller values indicate better effectiveness of the heuristic). Each color corresponds to a different hiding heuristic, each line type corresponds to a different way of selecting the edges in H. Colored areas (very narrow in most plots) represent the 95%-confidence intervals.

mum. On the other hand, when hiding random edges or edges from the vicinity of a node, the performance is further away from the optimum, particularly for Erdős-Rényi and Barabási-Albert networks. In all cases, the performance of our heuristics is significantly better than the random baseline.

Fig. 18 presents the results for the RSP problem. As can be seen, the performance of the Tally heuristic is at least 80% of the optimum when H contains edges with the highest scores. As is the case for the NSP problem, the performance compared to the optimum decreases when H is selected randomly or from the vicinity of the selected node, dropping to as low as about 15% in Erdős-Rényi networks.

8. Conclusions

In this paper, we introduced the problem of attacking sign prediction, whereby the aim of a network member is to hide the signs of a target set of links from a network analyst. This can be done by removing the signs of a predefined number of links. The computational analysis of this problem shows that it is NP-hard for both local and global similarity measures of sign prediction. In particular, it is NP-hard to delete a fixed number of signs of links in the network in order to:

- eliminate signed common neighborhood (and related local similarity measures) of the target links, i.e., make the signed common neighborhood of each link in the target set equal to zero, which means the analyst does not have any information on what the actual sign of any target link is,

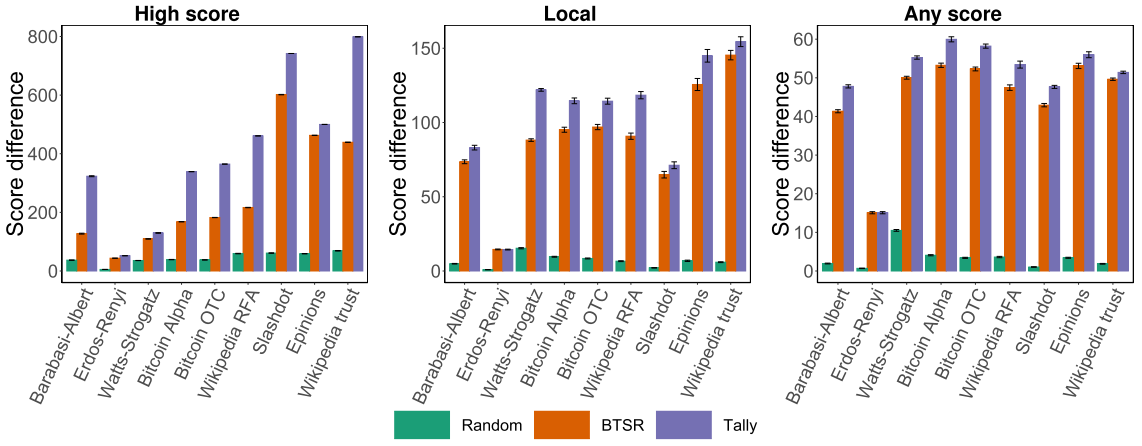


Fig. 13. The summed-up difference in the SCN values of the links in H after the hiding process for the RSP problem instances (larger values indicate better effectiveness of the heuristic). The left plot corresponds to H selected as the links with the greatest SCN scores, the middle plot to H selected as the links incident with the neighbors of a single node, while the right plot to links H selected uniformly at random. Error bars represent the 95%-confidence intervals.

- neutralize signed common neighborhood (and related local similarity measures) of the target links, i.e., make the signed common neighborhood of each link in the target set as close to zero as possible, which means the analyst has a minimal amount of information on what the actual sign of any target link is,
- neutralize global signed similarity measures, such as the Katz index,
- neutralize total signed common neighborhood (and related local similarity measures) of the target set, i.e., make the sum of signed common neighborhoods of each link in the target set as close to zero as possible, which minimizes the analyst's total information concerning the signs of links in the target set,
- reverse the signs of the links in the target set so that the analyst would have strictly inaccurate information concerning the signs of the links in the target set.

Nevertheless, we demonstrate several positive computational results, including an FPT-algorithm for eliminating signed common neighborhood (w.r.t. the predefined number of links signs of which can be removed by the network members). Furthermore, we show that if there exists a set W of important nodes in the network to the effect that the target set is the collection of all (and only) links between the members of this set W, then there is a polynomial-time algorithm for evading sign-prediction.

Finally, given the hardness results listed above, we proposed several heuristic algorithms for evading local similarity-based link prediction in signed networks. The theoretical results motivate the development of heuristics focused on *local* properties of the network, since the proofs demonstrate that analyzing the global features leads to computationally infeasible algorithms. The CTSR (Closed Triads Sign Removal) is based on removing the signs from edges belonging to a large number of closed triads containing edges the sign of which we wish to hide. Each such removal helps to obscure the signs of edges belonging to the closed triads. The BTSR (Balanced Triads Sign Removal) heuristic on the other hand, is inspired by the balance theory. It removes the signs in triads that are balanced, thereby helping to reverse the perceived sign of the selected edges. We also introduce the Tally versions of both heuristics which, by performing a small additional upkeep, is able to estimate not only the positive (from the point of view of the party running the heuristic) but also the negative consequences of each action. The experimental evaluation of the heuristics allows us to make some general observations. First, all of the proposed heuristics are vastly superior to their random baselines. Second, the Tally versions of the heuristics provide a modest, but not negligible, improvement over the basic versions of CTSR and BTSR. Third, the effectiveness of hiding in networks with varying size and density is highly dependent on the exact problem and type of network under consideration, without clear unifying trends. Finally, when compared to optimal solutions in smaller networks, our heuristic exhibit a performance falling within 80% to 100% of the optimum when hiding the signs of edges with high SCN scores, but the heuristics are much less effective when hiding the signs of edges selected uniformly at random.

There are many ways in which this work could be extended. Firstly, the attack model could be modified so that the network members could not only remove the signs of links, but also switch the signs or delete and introduce some links. Secondly, in our analysis, we focused on the evasion efforts by the network members. In the future, it would be interesting to consider the game-theoretic setting in which also the entity analyzing the networks is aware of the hiding attempts of the network members (in the spirit of, e.g., [48]). Furthermore, the present analysis builds on the similarity metrics that were proposed and analyzed in the literature as counterparts of a few similarity metrics for non-signed networks. There are, however, many other metrics for non-signed networks that could potentially work well if extended to signed networks. On top of this, one could also develop from scratch dedicated similarity measures for signed networks. Nevertheless, any analysis of this sort should focus first on demonstrating the usefulness and analyzing properties of newly extended/developed similarity metrics for signed networks before studying ways of evading them. Finally, the networks analyzed in the literature get increasingly

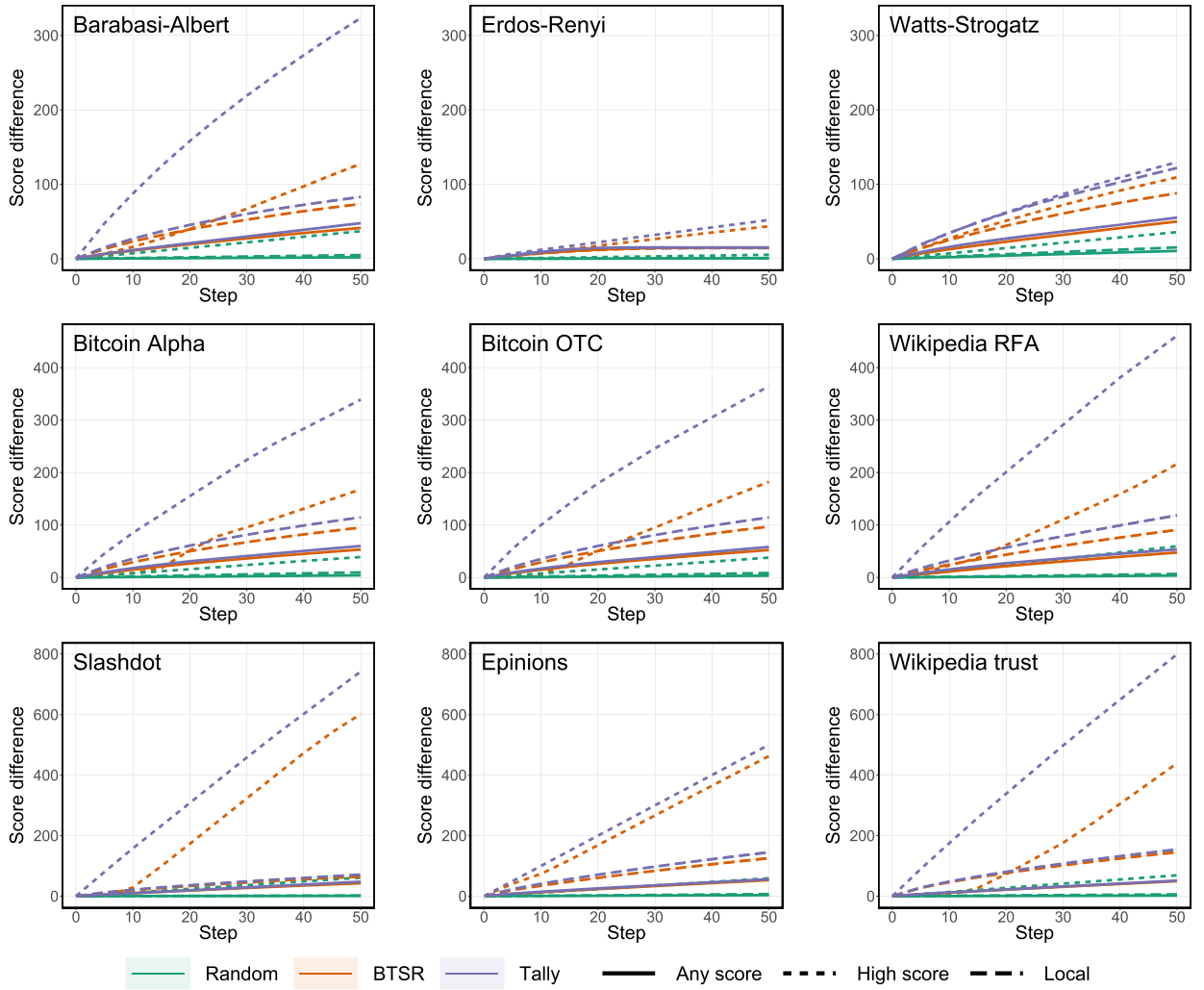


Fig. 14. A comparison of the heuristics' effectiveness for the RSP problem instances throughout the hiding process. In each plot, the x-axis represents the number of executions of the heuristic, while the y-axis represents the summed-up difference in SCN values of the links in H (larger values indicate better effectiveness of the heuristic). Each color corresponds to a different hiding heuristic, each line type corresponds to a different way of selecting the edges in H. Colored areas (very narrow in most plots) represent the 95%-confidence intervals.

more complex to keep up with the increasingly complex world (see, for instance, the abundance of types of multilayer networks [26]). In this vein, it could be interesting to extend our analysis in the future to more sophisticated versions of signed networks.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The public data information is provided in the paper

Acknowledgements

The research presented in this article was supported by the Polish National Science Centre (grant 2016/23/B/ST6/03599). This article is a significantly revised and extended version of the conference paper: [22]. Specifically:

- The following theoretical results have been added:

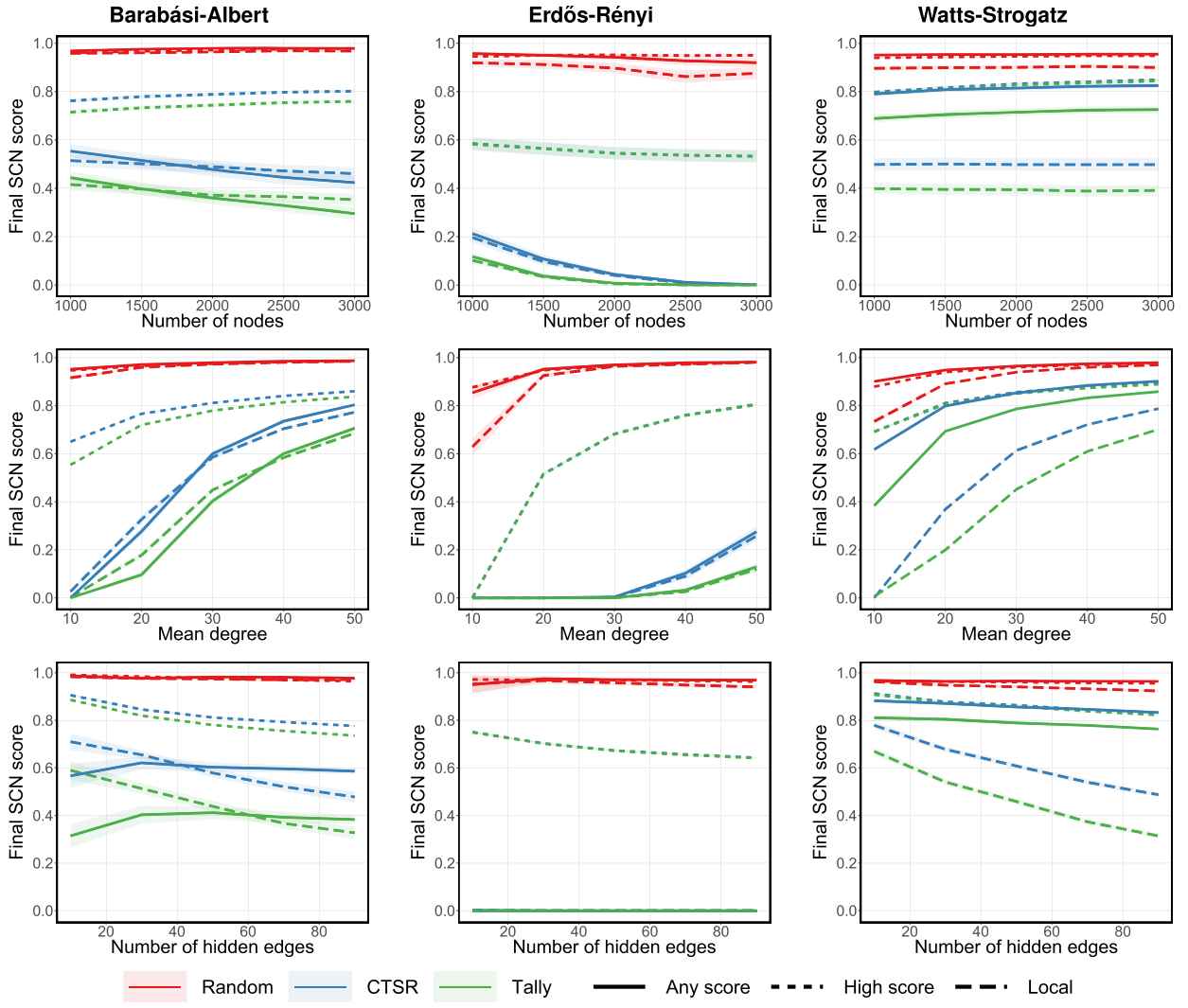


Fig. 15. A comparison of the performance of the hiding heuristics for NSP problem instances with varying characteristics. In each plot, the y-axis represents the relative sum of the absolute SCN scores of the links in H after the entire hiding process (smaller values indicate better effectiveness of the heuristic), while the x-axis represents the size of the network (top row), the mean degree of its nodes (mid row), or the size of H (bottom row). Each color corresponds to a different hiding heuristic, each line type corresponds to a different way of selecting the edges in H. Colored areas (very narrow in most plots) represent the 95%-confidence intervals.

  - Corollary 5 on the fW[2]-hardness of NSP w.r.t. the parameter k of the unrestricted version of NSP
  - Theorem 5 on the fixed-parameter tractability of ESCN w.r.t. the parameter k in the case when the goal of the attacker is to eliminate the signed common neighborhood for all pairs of nodes in the target set H to 0.
  - Theorem 6 on para-NP-hardness of ESCN w.r.t. the average degree of the input graph.
  - Theorem 10 on fixed-parameter tractability of NTSP with respect to the parameter $|H|$.
  - Theorem 8 on fixed-parameter tractability of NSCN with respect to the parameter $|H|$.
  - Corollary 6 on fixed-parameter tractability of ESCN with respect to the parameter $|H|$
- The proofs from the conference version were extended and revised. This also included adding explanatory figures to most of them.
- Finally, we extended the simulation results as follows:
  - While in the conference version we only studied real networks, we now consider synthetic networks (Barabasi-Albert, Erdos-Renyi, and Watts-Strogatz).
  - We also vary the parameterization of each type of the synthetic network in the robustness analysis.
  - We included a section on measuring the effectiveness of the Tally Heuristic against a stronger benchmark.

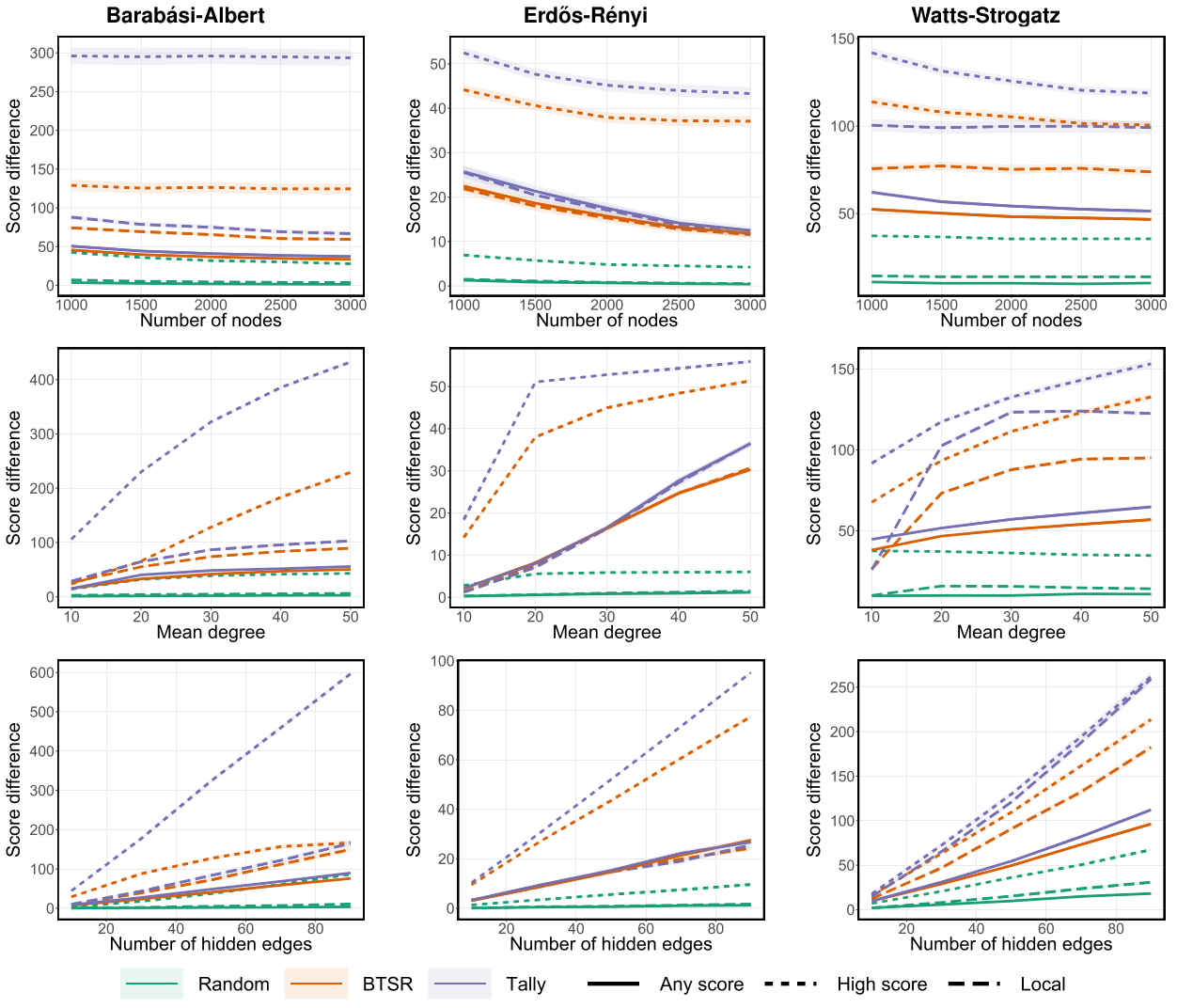


Fig. 16. A comparison of the performance of the hiding heuristics for RSP problem instances with varying characteristics. In each plot, the y-axis represents as the summed-up difference in SCN values of the links in H (larger values indicate better effectiveness of the heuristic), while the x-axis represents the size of the network (top row), the mean degree of its nodes (mid row), or the size of H (bottom row). Each color corresponds to a different hiding heuristic, each line type corresponds to a different way of selecting the edges in H. Colored areas (very narrow in most plots) represent the 95%-confidence intervals.

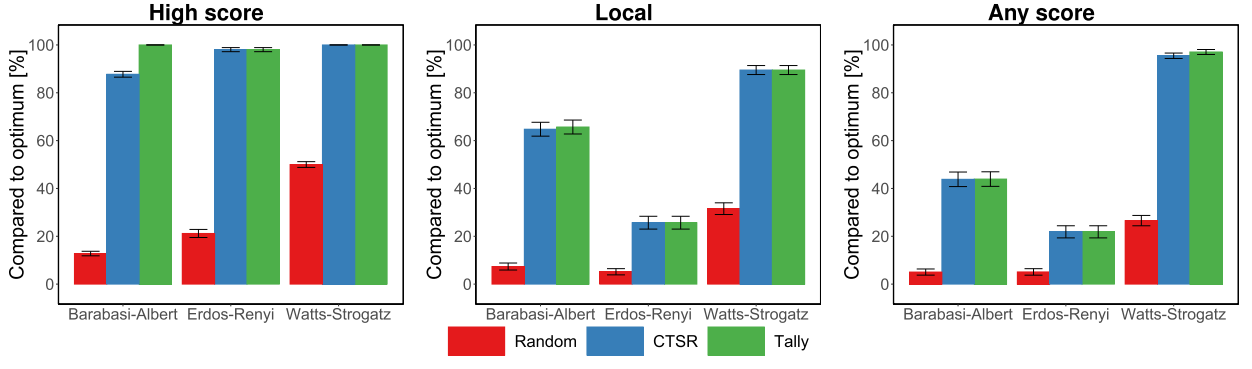


Fig. 17. The decrease in the sum of the absolute SCN scores of the links in H after the entire hiding process for the NSP problem instances in networks with 500 nodes, presented as the percentage of the optimal performance. The left plot corresponds to H selected as the links with the greatest SCN scores, the middle plot to H selected as the links incident with the neighbors of a single node, while the right plot to H selected uniformly at random. Error bars represent the 95%-confidence intervals. Scales are fixed to facilitate the comparison across subfigures.

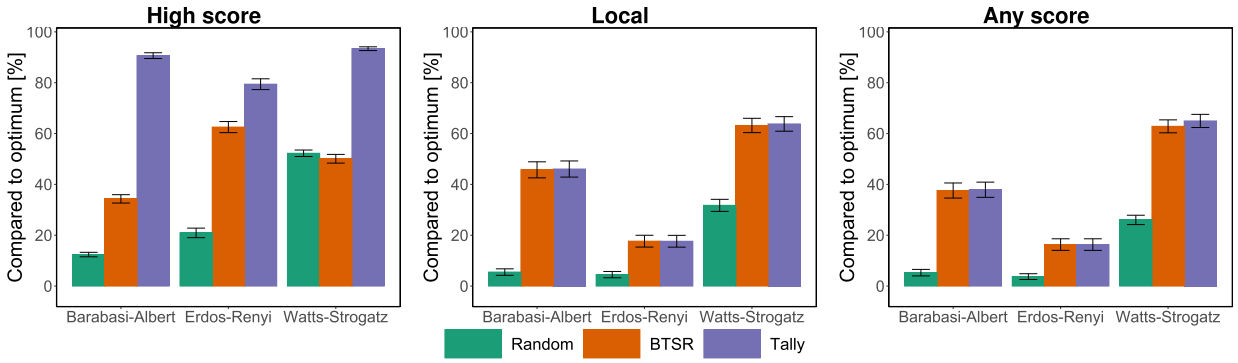


Fig. 18. The summed-up difference in the SCN values of the links in H after the hiding process for the RSP problem instances in networks with 500 nodes, presented as the percentage of the optimal performance. The left plot corresponds to H selected as the links with the greatest SCN scores, the middle plot to H selected as the links incident with the neighbors of a single node, while the right plot to H selected uniformly at random. Error bars represent the 95%-confidence intervals. Scales are fixed to facilitate the comparison across subfigures.

Appendix A. Summary of notation

Symbol	Description
$G = (V, E)$	The graph with the set of nodes V and the set of links E
$N(v)$	The set of neighbors of node v
$d(v)$	The degree of node v
$\|X\|$	The cardinality of a set X.
σ	Sign function assigning signs to links in E_G.
$\mathrm{CN}(u, v)$	The value of Common Neighborhood similarity measure for nodes u and v
$\mathrm{SCN}(u, v)$	The value of Signed Common Neighborhood similarity measure for nodes u and v
$\mathrm{J}(u, v)$	The value of Jaccard measure for nodes u and v
$\mathrm{SJ}(u, v)$	The value of Signed Jaccard similarity measure for nodes u and v
$\mathrm{PA}(u, v)$	The value of Preferential Attachment similarity measure for nodes u and v
$\mathrm{SPA}(u, v)$	The value of Signed Preferential Attachment similarity measure for nodes u and v
$\mathrm{K}(u, v)$	The value of Katz similarity measure for nodes u and v
$\mathrm{SK}(u, v)$	The value of Signed Katz similarity measure for nodes u and v
$N_+(u)$	Positive neighborhood of a node u.
$N_-(u)$	Negative neighborhood of a node u.
$d_+(u)$	Positive degree of a node u.
$d_-(u)$	Negative degree of a node u.
$c_s(u, v)$	Similar common neighborhood of nodes u and v.
$c_d(u, v)$	Dissimilar common neighborhood of nodes u and v.

References

[1] Priyanka Agrawal, Vikas K. Garg, Ramasuri Narayanam, Link label prediction in signed social networks, in: Twenty-Third International Joint Conference on Artificial Intelligence, 2013.
[2] Albert-László Barabási, Réka Albert, Emergence of scaling in random networks, Science 286 (5439) (1999) 509–512.
[3] Elisabetta Bergamini, Pierluigi Crescenzi, Gianlorenzo D'angelo, Henning Meyerhenke, Lorenzo Severini, Yllka Velaj, Improving the betweenness centrality of a node by adding links, ACM J. Exp. Algorithmics 23 (2018) 1–5.
[4] Moira Burke, Robert Kraut, Mopping up: modeling Wikipedia promotion decisions, in: Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work, CSCW '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 27–36.
[5] Dorwin Cartwright, Frank Harary, Structural balance: a generalization of Heider's theory, Psychol. Rev. 63 (5) (1956) 277.
[6] Jianer Chen, Iyad A. Kanj, Ge Xia, Improved paramterized upper bounds for vertex cover, in: Proc. 31st Mathematical Foundations of Computer Sicence 2006, 31st International Symposium, MFCS, 2006, pp. 238–249.
[7] Jinyin Chen, Lihong Chen, Yixian Chen, Minghao Zhao, Shanqing Yu, Qi Xuan, Xiaoniu Yang, Ga-based q-attack on community detection, IEEE Trans. Comput. Soc. Syst. 6 (3) (2019) 491–503.
[8] X. Chen, J.F. Guo, X. Pan, et al., Link prediction in signed networks based on connection degree, J. Ambient Intell. Humaniz. Comput. 10 (2019) 1747–1757.
[9] Xiao Chen, Jing-Feng Guo, Xiao Pan, Chunying Zhang, Link prediction in signed networks based on connection degree, J. Ambient Intell. Humaniz. Comput. 10 (5) (2019) 1747–1757.
[10] Liang Chen, Jintang Li, Jiaying Peng, Tao Xie, Zengxu Cao, Kun Xu, Xiangnan He, Zibin Zheng, A survey of adversarial learning on graphs, 2020, arXiv.
[11] Kewei Cheng, Jundong Li, Jiliang Tang, Huan Liu, Unsupervised sentiment analysis with signed social networks, in: 31st AAAI Conference on Artificial Intelligence, AAAI 2017, in: 31st AAAI Conference on Artificial Intelligence, AAAI Press, 2017, pp. 3429–3435, AAAI, 2017, conference date: 04-02-2017 through 10-02-2017.
[12] Kai-Yang Chiang, Cho-Jui Hsieh, Nagarajan Natarajan, Inderjit S. Dhillon, Ambuj Tewari, Prediction and clustering in signed networks: a local to global perspective, J. Mach. Learn. Res. 15 (1) (2014) 1177–1213.

[13] Pierluigi Crescenzi, Gianlorenzo d'Angelo, Lorenzo Severini, Yllka Velaj, Greedily improving our own centrality in a network, in: International Symposium on Experimental Algorithms, Springer, New York, USA, 2015, pp. 43–55.
[14] Pierluigi Crescenzi, Gianlorenzo D'angelo, Lorenzo Severini, Yllka Velaj, Greedily improving our own closeness centrality in a network, ACM Trans. Knowl. Discov. Data 11 (1) (2016) 9.
[15] Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Daniel Marx, Marcin Pilipczuk, Michał Pilipczuk, Saket Saurabh, Parametrized Algorithms, Springer, 2015.
[16] Tyler Derr, Chenxing Wang, Suhang Wang, Jiliang Tang, Signed node relevance measurements, arXiv preprint, arXiv:1710.07236, 2017.
[17] Palash Dey, Sourav Medya, Covert networks: how hard is it to hide?, in: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, IFAAMAS, Montreal, Canada, 2019, pp. 628–637.
[18] Palash Dey, Sourav Medya, Manipulating node similarity measures in networks, in: AAMAS '20: Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems, 2020, pp. 321–329.
[19] Paul Erdős, Alfréd Rényi, On random graphs I, Publ. Math. (Debr.) 6 (1959) 290–297.
[20] Valeria Fionda, Giuseppe Pirro, Community deception or: how to stop fearing community detection algorithms, IEEE Trans. Knowl. Data Eng. 30 (4) (2017) 660–673.
[21] M.R. Garey, D.S. Johnson, Computer and Intractability: a Guide to the Theory of NP-Completeness. A Series of Books in the Mathematical Sciences, W.H. Freeman & Co Ltd, 1979.
[22] Michał Tomasz Godziszewski, Tomasz P. Michalak, Marcin Waniek, Talal Rahwan, Kai Zhou, Yulin Zhu, Attacking similarity-based sign prediction, in: Proceedings of the 2021 IEEE International Conference on Data Mining (ICDM), IEEE Computer Society, 2021, pp. 1367–1372.
[23] R. Guha, Ravi Kumar, Prabhakar Raghavan, Andrew Tomkins, Propagation of trust and distrust, in: Proceedings of the 13th International Conference on World Wide Web, WWW '04, Association for Computing Machinery, New York, NY, USA, 2004, pp. 403–412.
[24] Baofang Hu, Hong Wang, Xiaomei Yu, Weihua Yuan, Tianwen He, Sparse network embedding for community detection and sign prediction in signed social networks, J. Ambient Intell. Humaniz. Comput. 10 (1) (2019) 175–186.
[25] Bert Huang, Angelika Kimmig, Lise Getoor, Jennifer Golbeck, A flexible framework for probabilistic models of social trust, in: Proceedings of the 6th International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction, SBP'13, Springer-Verlag, Berlin, Heidelberg, 2013, pp. 265–273.
[26] Mikko Kivelä, Alex Arenas, Marc Barthelemy, James P. Gleeson, Yamir Moreno, Mason A. Porter, Multilayer networks, J. Complex Netw. 2 (3) (2014) 203–271.
[27] Srijan Kumar, Francesca Spezzano, V.S. Subrahmanian, Accurately detecting trolls in slashdot zoo via decluttering, in: Xindong Wu, Martin Ester, Guandong Xu (Eds.), 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2014, Beijing, China, August 17-20, 2014, IEEE Computer Society, 2014, pp. 188–195.
[28] Srijan Kumar, Francesca Spezzano, V.S. Subrahmanian, Christos Faloutsos, Edge weight prediction in weighted signed networks, in: 2016 IEEE 16th International Conference on Data Mining (ICDM), IEEE, 2016, pp. 221–230.
[29] Jérôme Kunegis, Andreas Lommatzsch, Christian Bauckhage, The slashdot zoo: mining a social network with negative edges, in: Proceedings of the 18th International Conference on World Wide Web, 2009, pp. 741–750.
[30] H.W. Lenstra, Integer programming with a fixed number of variables, Math. Oper. Res. 8 (4) (1983) 538–548.
[31] Jure Leskovec, Daniel Huttenlocher, Jon Kleinberg, Predicting positive and negative links in online social networks, in: Proceedings of the 19th International Conference on World Wide Web, 2010, pp. 641–650.
[32] Jure Leskovec, Daniel Huttenlocher, Jon Kleinberg, Signed networks in social media, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2010, pp. 1361–1370.
[33] Dong Li, Zhi-Ming Xu, Nilanjan Chakraborty, Anika Gupta, Katia Sycara, Sheng Li, Sergio Gómez, Polarity related influence maximization in signed social networks, PLoS ONE 9 (2014) 7.
[34] Lü Linyuan, Zhou Tao, Link prediction in complex networks: a survey, Phys. A, Stat. Mech. Appl. 390 (6) (2011) 1150–1170.
[35] Silviu Maniu, Bogdan Cautis, Talel Abdessalem, Building a signed network from interactions in Wikipedia, in: Databases and Social Networks, Association for Computing Machinery, 2011, pp. 19–24.
[36] Alan Mislove, Bimal Viswanath, Krishna P. Gummadi, Peter Druschel, You are who you know: inferring user profiles in online social networks, in: Proceedings of the Third ACM International Conference on Web Search and Data Mining, 2010, pp. 251–260.
[37] Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, Noah A. Smith, From tweets to polls: linking text sentiment to public opinion time series, in: William W. Cohen, Samuel Gosling (Eds.), Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23-26, 2010, The AAAI Press, 2010, pp. 23–26.
[38] Emre Sarigol, David Garcia, Frank Schweitzer, Online privacy as a collective phenomenon, in: Proceedings of the Second ACM Conference on Online Social Networks, ACM, New York, USA, 2014, pp. 95–106.
[39] Mahsa Shafaei, Mahdi Jalili, Community structure and information cascade in signed networks, New Gener. Comput. 32 (2014) 08.
[40] Moshen Shahriari, Mahdi Jalili, Ranking nodes in signed social networks, Soc. Netw. Anal. Min. 4 (2014) 12.
[41] Konstantinos Sotiropoulos, John W. Byers, Polyvios Pratikakis, Charalampos E. Tsourakakis, Twittermancer: predicting user interactions on Twitter, in: 2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton), IEEE Press, 2019, pp. 973–980.
[42] Jiliang Tang, Yi Chang, Charu Aggarwal, Huan Liu, A survey of signed network mining in social media, ACM Comput. Surv. 49 (3) (2016) 1–37.
[43] V.A. Traag, Jeroen Bruggeman, Community detection in networks with positive and negative links, Phys. Rev. E 80 (2009) 036115.
[44] Marcin Waniek, Tomasz P. Michalak, Michael J. Wooldridge, Talal Rahwan, Hiding individuals and communities in a social network, Nat. Hum. Behav. 2 (2) (2018) 139.
[45] Marcin Waniek, Kai Zhou, Yevgeniy Vorobeychik, Esteban Moro, Tomasz P. Michalak, Talal Rahwan, How to hide one's relationships from link prediction algorithms, Sci. Rep. 9 (1) (2019) 1–10.
[46] Marcin Waniek, Tomasz Michalak, Talal Rahwan, Hiding in multilayer networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, New York, USA, vol. 34, AAAI, 2020, pp. 1021–1028.
[47] Marcin Waniek, Manuel Cebrian, Petter Holme, Talal Rahwan, Social diffusion sources can escape detection, 2021.
[48] Marcin Waniek, Jan Woźnica, Kai Zhou, Yevgeniy Vorobeychik, Talal Rahwan, Tomasz P. Michalak, Strategic evasion of centrality measures, in: Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems, IFAAMAS, UK, 2021, pp. 1389–1397.
[49] Tomasz Was, Marcin Waniek, Talal Rahwan, Tomasz Michalak, The manipulability of centrality measures-an axiomatic approach, in: Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS, Auckland, New Zealand, 2020, pp. 1467–1475.
[50] Duncan J. Watts, Steven H. Strogatz, Collective dynamics of small-world networks, Nature 393 (6684) (1998) 440–442.
[51] Robert West, Hristo S. Paskov, Jure Leskovec, Christopher Potts, Exploiting social network structure for person-to-person sentiment analysis, Trans. Assoc. Comput. Linguist. 2 (2014) 297–310.
[52] Robert West, Hristo S. Paskov, Jure Leskovec, Christopher Potts, Exploiting social network structure for person-to-person sentiment analysis, Trans. Assoc. Comput. Linguist. 2 (2014) 297–310.
[53] Zhaoming Wu, Charu C. Aggarwal, Jimeng Sun, The troll-trust model for ranking in signed networks, in: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, WSDM '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 447–456.

[54] Shuang-Hong Yang, Alexander J. Smola, Bo Long, Hongyuan Zha, Yi Chang, Friend or frenemy? Predicting signed ties in social networks, in: Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '12, Association for Computing Machinery, New York, NY, USA, 2012, pp. 555–564.

[55] Shanqing Yu, Minghao Zhao, Chenbo Fu, Jun Zheng, Huimin Huang, Xincheng Shu, Qi Xuan, Guanrong Chen, Target defense against link-prediction-based attacks via evolutionary perturbations, IEEE Trans. Knowl. Data Eng. 33 (2) (2019) 754–767.

[56] Weiwei Yuan, Chenliang Li, Guangjie Han, Donghai Guan, Li Zhou, Kangya He, Negative sign prediction for signed social networks, Future Gener. Comput. Syst. 93 (2019) 962–970.

[57] Kai Zhou, Tomasz P. Michalak, Yevgeniy Vorobeychik, Adversarial robustness of similarity-based link prediction, in: 2019 IEEE International Conference on Data Mining (ICDM), IEEE, 2019, pp. 926–935.

[58] Kai Zhou, Tomasz P. Michalak, Marcin Waniek, Talal Rahwan, Yevgeniy Vorobeychik, Attacking similarity-based link prediction in social networks, in: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, 2019, pp. 305–313.