

Optimal Surveillance of Covert Networks by Minimizing Inverse Geodesic Length

Serge Gaspers, Kamran Najeebullah

UNSW Sydney, Australia, and Data61, CSIRO, Australia
 sergeg@cse.unsw.edu.au, kamran.najeebullah@data61.csiro.au

Abstract

The inverse geodesic length (IGL) is a well-known and widely used measure of network performance. It equals the sum of the inverse distances of all pairs of vertices. In network analysis, IGL of a network is often used to assess and evaluate how well heuristics perform in strengthening or weakening a network. We consider the edge-deletion problem MINIGL-ED. Formally, given a graph G , a budget k , and a target inverse geodesic length T , the question is whether there exists a subset of edges X with $|X| \leq ck$, such that the inverse geodesic length of $G - X$ is at most T .

In this paper, we design algorithms and study the complexity of MINIGL-ED. We show that it is NP-complete and cannot be solved in subexponential time even when restricted to bipartite or split graphs assuming the Exponential Time Hypothesis. In terms of parameterized complexity, we consider the problem with respect to various parameters. We show that MINIGL-ED is fixed-parameter tractable for parameter T and vertex cover by modeling the problem as an integer quadratic program. We also provide FPT algorithms parameterized by twin cover and neighborhood diversity combined with the deletion budget k . On the negative side we show that MINIGL-ED is W[1]-hard for parameter tree-width.

1 Introduction

Network analysis, in particular the strategic aspect of network analysis, has attained profound interest in recent years due to its relevance across many fields of research, including Artificial Intelligence (e.g., (Aziz, Gaspers, and Najeebullah 2017; Michalak, Rahwan, and Wooldridge 2017)). One fundamental problem within this area is the identification of most critical vertices and edges of a network (Michalak et al. 2013; Chen et al. 2012; Zheng, Dunagan, and Kapoor 2011). The problem enjoys a vast area of applications including containing the spread of an epidemic (see, e.g., (Kovács and Barabási 2015)), weakening a terrorist network (see, e.g., (Aziz, Gaspers, and Najeebullah 2017)), preventing the spread of a contagion in a computer network (see, e.g., (Kuhlman et al. 2013)) and blocking rumors or "fake news" in social networks (see, e.g., (Zhang et al. 2016)).

We focus on the problem of weakening a covert network by identifying and deleting or monitoring the links that are

critical for a high performance of the network. The problem of weakening a covert network has mostly been considered with respect to vertex deletion (see, e.g., (Michalak et al. 2013; Aziz, Gaspers, and Najeebullah 2017; Aziz et al. 2018; Gaspers and Lau 2018)). A common trend here – seemingly influenced by social network analysis and epidemiology (see, e.g., (Carley, Reminga, and Kamneva 2003; Valente 2012) and (Kovács and Barabási 2015; Zhang et al. 2012)) – is to rank the vertices with respect to their importance and eliminate them in the order of maximum score (Michalak et al. 2013; 2015; Szczepanski, Michalak, and Rahwan 2015)). While the assumption that a node can be eliminated sounds realistic for social and epidemic networks, it might not be as practical for covert networks due to various geopolitical reasons.

Quantifying the performance of a network has been a topic of interest in network analysis, and several measures have been proposed. Some of the most frequently used measures of network performance are *component order connectivity* (size of the largest connected component) (Gross et al. 2013; Drange, Dregi, and van't Hof 2014), *clustering coefficient* (probability that two nodes are connected if both are connected to a common third node) (Watts and Strogatz 1998) and *inverse geodesic length (IGL)*. We choose to quantify the network performance by IGL. Formally, $IGL(G) = \sum_{\{u,v\} \subseteq V} \frac{1}{dist(u,v)}$. Our choice is dictated by the frequent use of IGL as a measure of network performance across various fields, such as AI (e.g. (Aziz, Gaspers, and Najeebullah 2017)), network security (e.g. (Holme et al. 2002)), social networks (e.g. (Morone and Makse 2015)) and game theory (e.g. (Holme et al. 2002; Michalak et al. 2015; Szczepanski, Michalak, and Rahwan 2015)). Moreover, Latora and Marchiori (2001) found IGL to be effective on small-world graphs and studied several networks systems to show that it is the underlying general principle of construction for several real-world networks including transportation, communication and neural networks.

A practical example to demonstrate how a higher IGL value indicates better network performance is of a communication network where the quality of the signal degrades when the distance between the nodes increases. We consider the problem of minimizing the IGL of a graph by allowing a limited number of edge deletions. One interpretation in this case is that the deletion of edges corresponds to surveillance

of communication channels.

MINIMIZE IGL BY EDGE DELETION (MINIGL-ED)

Input: Graph G , integer k , rational number T .
 Question: Does there exist $X \subseteq E(G)$, such that $|X| \leq k$ and $IGL(G - X) \leq T$?

We perform an extensive complexity analysis of the problem. We show that MINIGL-ED is NP-complete and cannot be solved in subexponential time even when restricted to bipartite and split graphs under standard assumptions. In terms of parameterized complexity we show that the problem is $W[1]$ -hard with respect to parameter tree-width. On the positive side we provide FPT algorithms for MINIGL-ED with respect to parameter target inverse geodesic length T and vertex cover τ . We also provide FPT algorithms parameterized by twin cover number and neighborhood diversity when combined with the deletion budget k .

Our choice of parameters is motivated by real-world datasets (see Table 1) and follows the trends of the existing literature. Both tree-width and vertex cover number are among the most widely studied structural parameters (Fellows et al. 2008; Fomin et al. 2014) while neighborhood diversity (Lampis 2012; Ganian 2012) and twin cover (Ganian 2015) are less restrictive generalizations of vertex cover that are more appropriate for dense graphs.

Overall, we note that MINIGL-ED is computationally hard even on restricted classes where many NP-complete problems can be solved in polynomial time. However, certain parameterizations of the problem provide algorithms of practical interest.

2 Preliminaries

We consider finite, undirected, simple graphs denoted as $G = (V, E)$. We denote the set of vertices and edges of G as $V(G)$ and $E(G)$ respectively, with $n = |V(G)|$ and $m = |E(G)|$. For any graph terminologies or notations used but not defined here we refer to (Diestel 2010). Let $u, v \in V(G)$ with $u \neq v$. The *distance* between u and v in G , denoted by $dist_G(u, v)$ (shorthand $dist(u, v)$), is the number of edges on a shortest path between u and v . The i th *neighborhood* of a vertex v is the set of vertices at distance i from v and is denoted by $N^i(v)$. We denote $N(v) = N^1(v)$. A pair of vertices at finite distance is *connected*.

Let $S \subseteq V(G)$. The graph induced on a vertex set S is denoted as $G[S]$ i.e., $V(G[S]) = S$ and $E(G[S]) = \{e \in E : e \subseteq S\}$. We denote by $G - S'$ the graph that is obtained by deleting all the edge in edge set S' from G . The IGL impact IGL_{imp} of an edge e is given by $IGL_{imp}(e) = IGL(G) - IGL(G - \{e\})$. We also utilize this notion for sets of edges. A d -*regular graph* is a graph where each vertex has d neighbors. A *bipartite graph* is a graph whose vertex set can be partitioned into two independent sets. A *split graph* is a graph whose vertex set can be partitioned into a clique \mathcal{K} and an independent set I . Such a partition (\mathcal{K}, I) is known as a split partition. Throughout this article we assume that \mathcal{K} is a maximal clique in a given split partition (\mathcal{K}, I) .

Two vertices u and v are *twins* if $N(u) \setminus \{v\} = N(v) \setminus \{u\}$. They are *true twins* if they are twins and $uv \in E(G)$,

Table 1: Covert networks (UCINET Software 2017) with number of vertices (n), edges (m), vertex cover number (τ), twin cover number (Γ), neighborhood diversity (η) and IGL.

| Dataset | n | m | τ | Γ | η | IGL |
|----------------|-----|------|--------|----------|--------|----------|
| FIFA | 450 | 5022 | 422 | 40 | 67 | 36500.67 |
| Drug Net | 293 | 192 | 76 | 63 | 147 | 3497.22 |
| 9/11 Hijackers | 61 | 124 | 27 | 26 | 52 | 639.05 |
| Siren | 44 | 103 | 18 | 9 | 21 | 475.17 |
| Cocaine Jake | 38 | 50 | 8 | 8 | 19 | 295.08 |
| Montreal Gangs | 35 | 78 | 12 | 12 | 29 | 801.32 |
| Togo | 33 | 47 | 10 | 7 | 20 | 242.0 |
| Cielnet | 25 | 35 | 7 | 7 | 15 | 141.03 |
| Greece | 18 | 30 | 6 | 6 | 5 | 37.50 |

the edge uv is called a *twin edge*. A set of vertices $C \subseteq V(G)$ is a *twin cover* of G , if for every edge $xy \in E(G)$, either xy is incident to a vertex in C or xy is a twin edge (Ganian 2015). Given a graph G and an integer k , the TWIN COVER problem is to determine whether there exists a twin cover of G of size at most k (Ganian 2015). We denote the twin cover number of a graph by Γ (i.e., $\Gamma = |C|$). A graph G has neighborhood diversity η , if there exists a partition of $V(G)$ into at most η sets, such that the vertices in each set are twins, such a partition is called the neighborhood partition of G and can be computed in polynomial time (Lampis 2012).

Let Π be a parameterized decision problem. We say Π is in *FPT (Fixed Parameter Tractable)*, if there is an algorithm solving any instance x with parameter k in time $f(k) \cdot |x|^c$, where $f(k)$ is an arbitrary function of k and c is a constant. The class $W[1]$ is a class of parameterized decision problems closed under so-called parameterized reductions. Any $W[1]$ -hard problem is unlikely to have an FPT algorithm. We refer to (Downey and Fellows 2013) for a detailed exposition of parameterized complexity. A *kernel*, or *kernelization* algorithm for a parameterized problem is a polynomial time algorithm producing an equivalent instance of the same parameterized problem such that the size of the resulting instance is upper bounded by a function of the input parameter. The *exponential time hypothesis* (ETH) is a conjecture coined by Impagliazzo and Paturi (2001) that implies that 3-SAT has no $2^{o(n')}$ -time algorithm, where n' is the number of variables in the input instance.

Proofs that are omitted or only sketched are deferred to the full version of the paper.

3 NP-hardness and ETH lower bounds

In this section we consider the classical computational complexity of MINIGL-ED. We show that MINIGL-ED is NP-complete and cannot be solved in subexponential time even when restricted to bipartite graphs or split graphs. Notice that the NP-completeness of MINIGL-ED will follow from Theorems 3.1 and 3.2. It also follows from a natural connection with the CLUSTER EDGE DELETION (CED) problem. CED is a well-known NP-complete problem (Shamir, Sharan, and Tsur 2004), where given a graph G and an integer k , the question is whether there exists a set of edges $X \subseteq E(G)$ of size at most k , such that $G - X$ is a dis-

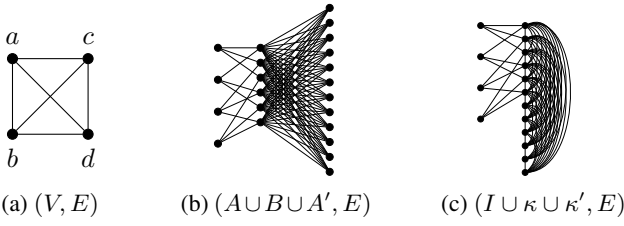


Figure 1: Figures (b) and (c) depict a bipartite and split graph obtained by transforming the 3-regular graph in (a).

joint union of cliques. Given an instance (G, k) of CED, we can obtain an equivalent instance $(G, k, T = m - k)$ of MINIGL-ED where $m = |E(G)|$.

In this section we show that MINIGL-ED remains NP-complete when restricted to bipartite graphs. Moreover, assuming that ETH holds, this bipartite restriction cannot be solved in time $2^{o(n)}$. We provide a reduction from the VERTEX COVER (VC) problem on 3-regular graphs. It is known that VC cannot be solved in time $2^{o(n+m)}$ even when restricted to 3-regular graphs (Sandeep and Sivadasan 2015) unless the ETH fails.

Theorem 3.1. *The restriction of MINIGL-ED to bipartite graphs is NP-complete and cannot be solved in time $2^{o(n)}$ unless the ETH fails.*

Proof. Let (G, k) be an instance of VERTEX COVER on 3-regular graphs. Without loss of generality, we assume that $k < n$ and therefore, $3k < 2m$. We construct an instance (G', k', T) for MINIGL-ED as follows: define $V(G') = A \cup A' \cup B$, where $A = V(G)$, $A' = \{v_e, v'_e : e \in E(G)\}$ and $B = E(G)$. Define $E(G') = \{v_1 v_2 : v_1 \in A', v_2 \in B\} \cup \{v_i v_e : v_i \in A, v_e \in B, v_i \in e\}$. See Figure 1b for a basic example of this construction. Set $k' = 3k$ and $T = T(k, 2m - 3k, 0)$ where;

$$\begin{aligned} T(\alpha_0, \beta_1, \beta_2) &= 2m^2 + \beta_1 + 2\beta_2 + \frac{1}{2}(2m(n - \alpha_0)) \\ &+ \frac{m(5m - 3)}{2} + \beta_2 + \frac{1}{3}(m(n - \alpha_0) - \beta_1 - 2\beta_2) \\ &+ \frac{1}{4}\left(\frac{(n - \alpha_0)(n - \alpha_0 - 1)}{2} - \beta_2\right). \end{aligned} \quad (1)$$

Here α_0 denotes the number of isolated vertices in A and β_i denotes the number of vertices in B with i neighbors in A . This concludes our construction. Before we move to the formal proof, we establish that in order to minimize $IGL(G')$ deleting edges between A and B is at least as good as deleting edges between A' and B .

Let $L = (A \times B) \cap E(G')$, $R = (A' \times B) \cap E(G')$ and $X \subseteq E(G')$ be the set of edges in the solution. Let us first assume that $|R \cap X| < 3$, notice that each such deletion only increases the distance between its end points from 1 to 3 and this decreases the IGL by $2/3$. We can replace each edge in $R \cap X$ with any edge in $L \setminus X$ as it decreases the IGL value by at least the same amount. Now suppose that $|R \cap X| \geq 3$. Again deleting each such edge decreases the IGL by $2/3$, unless we have a vertex $b \in B$ such that $|N(b) \cap A'| = 0$ or a vertex $a \in A'$ such that $|N(a)| = 0$. Since $|X| \leq 3k$, $3k < 2m$ and $|N(b') \cap A'| = 2m$ where $b' \in$

B , such a vertex b does not exist. Also note that each pair of vertices $b, b' \in B$ are connected to each other through $2m$ distinct paths of length 2, and as $3k < 2m$ the distance between any pair of vertices in B cannot be increased. This also implies that the distance between a pair of vertices in A cannot be increased by deleting edges in R . On the other hand, it requires at least m edge deletions to isolate a vertex in A' . Since $|X| \leq 3k$ and $3k < 2m$, we can isolate at most 1 vertex in A' , even if $|R \cap X| = 3k$. Observe that, isolating a vertex $a \in A'$ decreases the IGL by $m + \frac{1}{2}(m + n - 1)$. But then for each set of 3 edges in $R \cap X$ we can replace them with edges in $L \setminus X$ such that the number of isolated vertices in A is maximized. This way we can isolate k vertices in A . Since isolating each vertex in A decreases the IGL by at least $\frac{4}{3}m + \frac{1}{4}(n - k + 11)$, deleting m edges in L decreases the IGL by at least $\frac{m}{3}(\frac{4m}{3} + \frac{1}{4}(n - k + 11))$, which is at least as much as deleting any combination of edges in R .

We now show that (G', k', T) is a Yes-instance of MINIGL-ED iff (G, k) is a Yes-instance for VERTEX COVER. Suppose (G', k', T) is a Yes-instance and there exists a set $X' \subseteq E(G')$ such that $IGL(G' - X') \leq T$ and $|X'| = k'$. Without loss of generality, we assume that $X' \subseteq L$. Let $G'' = G' - X'$, $\gamma_0 = |\{x \in A : |N(x) \cap B| = 0\}|$, $\delta_1 = |\{x \in B : |N(x) \cap A| = 1\}|$ and $\delta_2 = |\{x \in B : |N(x) \cap A| = 2\}|$. We have, $IGL(G'') = T(\gamma_0, \delta_1, \delta_2)$. Let us now obtain the values of γ_0, δ_1 and δ_2 . In order to obtain $IGL(G'') \leq T$, we will show below that we need to delete edges such that $\gamma_0 \geq k$, $\delta_1 \leq 2m - 3k$ and $\delta_2 = 0$. In other words we need to choose X' such that at least k vertices are isolated in A . Since each vertex in A has degree 3 and $k' \leq 3k$, we can isolate a set C' of at most k vertices in A . Moreover, in order to obtain $\delta_2 = 0$, we need to isolate the vertices in C' such that there is no vertex in B with two neighbors in A . But then the isolated vertices in A form a vertex cover of G . Hence we have that (G, k) is a Yes-instance for VERTEX COVER.

It remains to show that removing k' edges from G' optimally gives an IGL of $T(k, 2m - 3k, 0)$. Given that m and n carry over from the original instance and are part of the input, we observe that in order to minimize the IGL, it is sufficient to first, maximize α_0 , and secondly minimize $T' = \frac{19\beta_2}{12} + \frac{2\beta_1}{3} + 0\beta_0$, where T' is obtained by restricting $T(\alpha_0, \beta_1, \beta_2)$ to the terms that only contain β_1 and/or β_2 . Notice that it takes one edge deletion to decrease β_2 and increase β_1 by 1, and it also takes one edge deletion to decrease β_1 and increase β_0 by 1. Moreover, the first kind of edge deletion decreases T' by $\beta_2 - \beta_1 = \frac{11}{12}$ and the second kind of edge deletion decreases T' by $\beta_1 - 0 = \frac{2}{3}$. This implies that it is optimal to prefer edge deletions that minimize the value of β_2 . Since $k' = 3k$, T' is minimized by performing edge deletions such that $\beta_2 = 0$, $\beta_1 = 2m - 3k$ and $\beta_0 = 3k - m$. On the other hand, since each vertex in A has degree 3, we have that $\alpha_0 \leq k$.

Conversely, suppose (G, k) is a Yes-instance for VERTEX COVER and there exists a vertex cover $X \subseteq V(G)$ such that $|X| = k$. Then by deleting edges incident to the vertices corresponding to X in G' , we isolate k vertices in A and there is no vertex in B with two neighbors in A . Thus we

have that (G', k', T) is a Yes-instance for MINIGL-ED. \square

We further obtain that MINIGL-ED remains NP-hard and cannot be solved in subexponential time (assuming the ETH), even when restricted to split graphs. This can be done by a reduction from VERTEX COVER on 3-regular graphs that is similar in spirit to the reduction in Theorem 3.1 and is depicted in Figure 1c.

Theorem 3.2. *The restriction of MINIGL-ED to split graphs is NP-complete and cannot be solved in time $2^{o(n)}$ unless the ETH fails.*

4 Parameterized Algorithms

Here we provide parameterized algorithms for MINIGL-ED with respect to several parameters. Notice that if $m > k + T$ then we have a No-instance for MINIGL-ED, as each edge contribute at least 1 to the IGL. Thus, MINIGL-ED has a kernel of size $O(k + T)$. Consequently, we have that MINIGL-ED is FPT for parameter $k + T$.

Next, we show that MINIGL-ED remains FPT when parameterized by T , even if the value of k is unbounded.

4.1 Parameter T

We provide an FPT algorithm that given an instance (G, k, T) of MINIGL-ED in polynomial time, outputs a solution or concludes that G has a vertex cover of size at most $2T$. We will see in Theorem 4.4 that MINIGL-ED is FPT when parameterized by the vertex cover number. Our algorithm will rely on some reduction rules based on the following notions.

A *matching* M in a graph is a set of edges $M \subseteq E$ such that no two edges in M share a vertex. A matching M is said to be *maximum* if for any other matching M' , $|M| \geq |M'|$. A maximum matching can be computed in polynomial time (Micali and Vazirani 1980). Let M be a maximum matching of G . It is known that $G - V(M)$ is an independent set, where $V(M)$ denotes the set of vertices incident to the edges in M . Let $m' = |E(G)| - k$.

Our reduction rules are applied in the same order as they are defined here;

Reduction Rule 4.1. *If $m' > T$ return No.*

Reduction Rule 4.2. *If $|M| \geq m'$ return Yes.*

Reduction rule 4.1 is correct as a graph on m edges has IGL at least m . Consider reduction rule 4.2, if $|M| \geq m'$ then we have a set of edges $M' \subseteq M$, such that $|M'| = m'$. By deleting the set of k edges $E(G) \setminus M'$ in G we obtain, $IGL(G[V(M')]) = m'$. Thus, due to reduction rule 4.1, $IGL(G[V(M')]) = m' \leq T$. Hence (G, k, T) is a Yes-instance.

After applying reduction rules 4.1 and 4.2, G has a maximum matching M of size less than m' . Notice that $V(M)$ is a vertex cover of G where $|V(M)| \leq 2m' \leq 2T$. Thus;

Theorem 4.3. *MINIGL-ED is FPT for parameter T .*

We observe that the above argument also establishes that MINIGL-ED is FPT for parameter m' since the reduced instance has a vertex cover of size at most $2m'$.

4.2 Parameter τ

We now turn to structural parameters and consider the well-known parameter vertex cover number. Note that VERTEX COVER is NP-complete but FPT parameterized by the vertex cover number (Chen, Kanj, and Xia 2010). Therefore we may as well assume that a vertex cover of minimum size is provided as part of the input. It is known that in a graph G with vertex cover number τ , the diameter of each connected component in G is at most 2τ (Aziz et al. 2018). We provide an FPT algorithm for MINIGL-ED parameterized by vertex cover number using an *integer quadratic program (IQP)*. An IQP is an optimization problem whose input is an $\xi \times \xi$ integer matrix Q , $\mu \times \xi$ integer matrices A and W and μ -dimensional integer vectors b and c . The task is to solve the following optimization problem:

$$\begin{aligned} & \text{minimize} && y^{\text{tr}} Q y \\ & \text{subject to} && A y \leq b \\ & && W y = c \\ & && y \in \mathbb{Z}^{\xi}. \end{aligned}$$

Note that the constraints are linear, whereas the objective function can be quadratic.

Lokshantov (2015) showed that there exists an algorithm that given an instance of IQP, runs in time $f(\xi, \alpha)n^{O(1)}$ and outputs an optimal solution $y \in \mathbb{Z}^{\xi}$, where α is the maximum absolute value in A and Q .

Theorem 4.4. *MINIGL-ED is FPT for parameter vertex cover number.*

Proof. Let (G, k, T) be an instance of MINIGL-ED with $G = (V, E)$, and let C be a smallest vertex cover in G with $|C| = \tau$. To find a set of edges that need to be deleted in $E(G[C])$, we enumerate all subsets F of $E(G[C])$ and guess the set F' that need to be part of our solution. Since $|E(G[C])| \leq \tau^2$, we need to enumerate at most 2^{τ^2} subsets of edges. Set $C = C \setminus F$, $G = G - F$ and $k = k - |F|$.

Since C is a vertex cover, we have that $I = V(G) \setminus C$ is an independent set. We partition I into a set of equivalence classes \mathcal{P} such that any two vertices u, v are in the same equivalence class $P_i \in \mathcal{P}$, if $N(u) = N(v)$. Notice that $|\mathcal{P}| \leq 2^{\tau}$. Clearly, \mathcal{P} can be computed in time $O^*(2^{\tau})$.

Observe that when we delete an edge uv incident to a vertex $u \in P_i$, u moves from P_i to P_j , where $N(P_j) = N(P_i) \setminus \{v\}$. Similarly, if we delete at least one edge incident to each vertex in P_i , we have that $|P_i| = 0$. As each vertex in P_i moves to another equivalence class in \mathcal{P} and we say that P_i is *vanished*. Considering this observation we define a set of *vanishing* functions of the form $f : [2^{\tau}] \rightarrow \{0, 1\}$, where if a class P_i vanishes after deleting edges in X then $f(i) = 0$ otherwise $f(i) = 1$. The number of such functions is bounded by $2^{|\mathcal{P}|} = 2^{2^{\tau}}$, and all such functions f can be constructed and enumerated in the same time. For shorthand we denote $f(i)$ by f_i .

Observe that it might take more than one edge deletion to move a vertex $u \in P_i$ to P_j , where $P_i, P_j \in \mathcal{P}$. To adequately describe this, we introduce a function $\mu : [2^{\tau}], [2^{\tau}] \rightarrow \mathbb{Z}$, where $\mu(i, j) = 0$ if either $i = j$ or

$N(P_j) \not\subseteq N(P_i)$. This means that a vertex from class P_i can never move to P_j . On the other hand, if $i \neq j$ and $N(P_j) \subseteq N(P_i)$ then $\mu(i, j) = |N(P_i)| - |N(P_j)|$. This denotes the number of edges incident to a vertex in P_i that need to be deleted to move it to P_j . For shorthand, we will denote $\mu(i, j)$ by $\mu_{i,j}$.

We now construct an IQP for computing the minimum IGL of a graph G with a vertex cover C , equivalence classes \mathcal{P} and functions f and μ . Let variables x_i and $y_{i,j}$ respectively represent the number of vertices in each equivalence class P_i and the number of vertices that move from P_i to P_j , after deleting the edges in an optimal solution X .

We consider the following integer quadratic program that finds the optimal edge set X that minimizes $IGL(G - X)$.

$$\begin{aligned}
& \text{minimize} && OBJ \\
& \text{subject to} && x_i = 0 && 0 \leq i \leq 2^\tau, f_i = 0 \\
& && y_{i,j} = 0 && 0 \leq i, j \leq 2^\tau, \mu_{i,j} = 0 \\
& && 0 \leq y_{i,j} \leq k && 0 \leq i, j \leq 2^\tau, \mu_{i,j} > 0 \\
& && x_i = |P_i| + \sum_{j \in [2^\tau]} (y_{j,i} - y_{i,j}) && 0 \leq i \leq 2^\tau \\
& && \sum_{i \in [2^\tau]} x_i = n - \tau \\
& && \sum_{i,j \in [2^\tau]} y_{i,j} \cdot \mu_{i,j} \leq k,
\end{aligned}$$

where OBJ is a function, defined below, which represents the IGL of G after deleting the edges in X as defined by the variables x_i and $y_{i,j}$.

Since the function f determines which equivalence classes vanish, we can compute the distances between every pair of non-vanishing equivalence classes and vertex cover vertices. This is exactly the distance in the graph G' obtained from G by deleting all vanishing equivalence classes and merging each remaining equivalence class into a single vertex. We denote by $\delta(x, y)$ the distance between x and y in G' (x and y are either vertices from C or equivalence classes from P represented by vertices). We have that $\delta(P_i, P_i) = \infty$ if $N(P_i) = \emptyset$ and $\delta(P_i, P_i) = 2$ otherwise.

We now define OBJ' , which is the inverse geodesic length of the graph obtained after deleting X :

$$\begin{aligned}
OBJ' := & \sum_{u \in C} \sum_{v \in C \setminus \{u\}} \frac{1}{\delta(u, v)} + \sum_{u \in C} \sum_{\substack{i \in [2^\tau] \\ f(i)=1}} \frac{x_i}{\delta(u, P_i)} \\
& + \sum_{\substack{i \in [2^\tau] \\ f(i)=1}} \frac{x_i \cdot (x_i - 1)}{2 \cdot \delta(P_i, P_i)} + \sum_{\substack{i \in [2^\tau] \\ f(i)=1}} \sum_{\substack{j \in [2^\tau] \\ f(j)=1}} \frac{x_i \cdot x_j}{\delta(P_i, P_j)}
\end{aligned}$$

Since the coefficients of OBJ need to be integers, we define OBJ by multiplying OBJ' by the least common multiple of the integers 1 to 2^τ . We observe that all coefficients that multiply the variables x_i are bounded by a function of τ .

Enumerating all subsets of the edge set of the original vertex cover requires $O(2^{2^\tau})$ running time, and there are at most 2^{2^τ} possible vanishing functions f , resulting in at most $2^{2^\tau} \cdot 2^{2^\tau}$ calls to the IQP. The IQP itself however contains $2^\tau + 2^\tau \cdot 2^\tau$ variables, with a maximum integer coefficient upper bounded by a function of τ . Thus, due to the result of (Lokshtanov 2015), we obtain that our formulated IQPs are all FPT with respect to τ . Hence, MINIGL-ED is FPT for parameter vertex cover number (τ). \square

4.3 Parameter $\Gamma + k$

In this section we consider the twin cover number of the graph along with the deletion budget k as a parameter. The TWIN COVER problem is known to be NP-complete (Ganian 2015). However, it is fixed parameter tractable parameterized by the size of the twin cover and can be computed in time $O(|E| + \Gamma \cdot |V| + 1.2738^\Gamma)$ (Ganian 2015). Therefore, it is safe to assume that the input contains a smallest twin cover of the input graph.

Notions Let (G, k, T) be an instance of MINIGL-ED and C be a twin cover of G with $|C| = \Gamma$. Let \mathcal{P}_c be a partition of $V(G) \setminus C$ into a set of equivalence classes $\{P_1, P_2, \dots, P_q\}$, such that, for any two vertices $u, v \in V(G) \setminus C$, if $N(u) \cap C = N(v) \cap C$, then u and v belong to the same equivalence class. There are at most 2^Γ equivalence classes in \mathcal{P}_c one corresponding to each subset of C . Ganian (2015) noted that $G[V(G) \setminus C]$ is a disjoint union of cliques. In particular, for each $P_i \in \mathcal{P}_c$, $G[P_i]$ is a disjoint union of cliques. Let $\{\mathcal{P}_s, \mathcal{P}_l\}$ be a partition of the equivalence classes in \mathcal{P}_c where $\mathcal{P}_s = \{P_a \in \mathcal{P}_c : |P_a| \leq k\}$ and $\mathcal{P}_l = \mathcal{P}_c \setminus \mathcal{P}_s$. Denote by $Z_k(P_i)$ and $Z_l(P_i)$ the set of cliques of size at most k and at least $k + 1$ in P_i , respectively. Denote $V_s = \bigcup_{P_i \in \mathcal{P}_s} P_i$, $V_{Z_k} = \bigcup_{P_i \in \mathcal{P}_l} V(Z_k(P_i))$, $E_i(P_i) = \{uv \mid \{u, v\} \cap V(P_i) \neq \emptyset\}$, $E_s = \{uv \mid \{u, v\} \cap V_s \neq \emptyset\}$, and $E_{Z_k} = \{uv \mid \{u, v\} \cap V_{Z_k} \neq \emptyset\}$. Lastly, denote $Z_{min}(P_i) = \min_{Z_i \in Z_l(P_i)} |Z_i|$ and $E_l(P_i) = \{uv \mid \{u, v\} \cap (P_i \setminus V(Z_k(P_i))) \neq \emptyset \text{ and } \{u, v\} \cap C \neq \emptyset\}$. In the description of the algorithm, when we state that the algorithm *guesses* an object, this means that the algorithm branches into all possible values that this object can take. Using these notations, we outline Algorithm 1.

Before we discuss about the correctness of the algorithm, we note that when we delete an edge $uv \in E(G)$ where $\{u, v\} \not\subseteq C$, C does not remain a valid twin cover of $G - uv$. However, a valid twin cover can be obtained by setting $C = \{u, v\} \cup C$. In other words, after each such edge deletion, the twin cover can grow in size by up to 2. Resultantly, after k deletions we may have a twin cover C of size $\Gamma + 2k$. Similarly, observe that each such edge deletion increases the number of equivalence classes by 1. Consequently, after k edge deletions, we may have $2^\Gamma + k$ equivalence classes.

Correctness Clearly, Steps 1, 4 and 5 are exhaustive while Steps 2, 3, 7, 10 and 11 are trivial. In Step 8, for each equivalence class $P_i \in \mathcal{P}_l$ the algorithm picks $k^* \leq k$ edges that need to be removed from $E_{Z_k}(P_i)$. Although the size of the cliques in $Z_k(P_i)$ is bounded by k , the number of such cliques is unbounded. However, as we can delete at most k edges, we can only consider k cliques each of size $1, 2, \dots, k$. Hence the set of edges we consider for deletion is $E'_{Z_k}(P_i) \subseteq E_{Z_k}(P_i)$, where $|E'_{Z_k}(P_i)| \leq k((\Gamma + 2k)(1 + 2 + 3 + \dots + k) + (1 + 3 + 6 + \dots + \frac{k \cdot (k-1)}{2})) \leq O(k^7 \cdot \Gamma)$. We now establish the following results in order to prove the correctness of Step 9.

Lemma 4.4.1. *Given a graph G , a twin cover C of G , an equivalence class $P_i \in \mathcal{P}_l$ (as defined in Notions) and a pair of edges uv and xy where $\{uv, xy\} \subseteq E_l(P_i)$ with $uv \cap C = \emptyset$ and $xy \cap C \neq \emptyset$, $IGL(G - xy) \leq IGL(G - uv)$.*

Proof. Since each clique $Z_i \in \mathcal{Z}_l(P_i)$ has size at least $k + 1$, deleting an edge $uv \in E(Z_i)$ only increase the distance between u and v from 1 to 2, while all other pairwise distances remain unchanged. On the other hand, by definition of twin cover for any two vertices $x_1, x_2 \in V(Z_i)$, $N(x_1) \cap C = N(x_2) \cap C$. Resultantly, deleting an edge xy where $x \in V(Z_i)$ and $y \in N(V(Z_i))$ increases the distance between x and y from 1 to 2. However it also increases any shortest path, which starts(or ends) at x and passes through y , by 1. Thus, $IGL(G - xy) \leq IGL(G - uv)$. \square

Given the above result, we know that in Step 9 the algorithm branches on each equivalence class $P_i \in \mathcal{P}_l$ to choose an equivalence class. For the chosen class P_i the algorithm deletes an edge of the form xy where $x \in P_i$ and $y \in C$. As $|C| \leq \Gamma + 2k$, y can be chosen exhaustively by branching over all neighbors of P_i . However, it remains to show, how the algorithm chooses x , as the number of cliques in $\mathcal{Z}_l(P_i)$ is unbounded. The following lemma shows that it is optimal to choose x greedily from $Z_{min}(P_i)$.

Lemma 4.4.2. *Given a graph G , a twin cover C of G , an equivalence class $P_i \in \mathcal{P}_l$ and a pair of edges wy and xy where $\{wy, xy\} \subseteq E_l(P_i)$ with $y \in C$, $w \in (P_i \setminus V(Z_{min}(P_i)))$ and $x \in V(Z_{min}(P_i))$, $IGL(G - xy) \leq IGL(G - wy)$.*

Proof. Suppose that $N(P_i) = \{y\}$. Since both w and x belong to the cliques of size at least $k + 1$, deleting the edges wy and xy increase the distance between the end points of the edges from 1 to 2 and increase the distance to the vertices that are connected to them via a shortest path through y by 1. However, since y is the only neighbor of P_i , the shortest path between each pair of vertices in $V(Z_l(P_i))$ except for those that belong to the same clique passes through y . This means that deleting an edge of the form vy , for $v \in V(Z_i)$ and $Z_i \in \mathcal{Z}_l(P_i)$, increases the distance between each pair of vertices uv where $u \in (V(Z_l(P_i)) \setminus V(Z_i))$ from 2 to 3. We know that, $|V(Z_l(P_i)) \setminus V(Z_{min}(P_i))| \geq |V(Z_l(P_i)) \setminus V(Z_j)|$ for all $Z_j \in \mathcal{Z}_l(P_i)$. Thus $IGL(G - xy) \leq IGL(G - wy)$. \square

Theorem 4.5. *MINIGL-ED can be solved in $O(\Gamma^{2k} \cdot (k(2^\Gamma + k)(\Gamma + 3k))^k \cdot k^{2^\Gamma + k} \cdot (k^7 \cdot \Gamma)^k \cdot (\Gamma + 2k)^k)$, where Γ is the neighborhood diversity of the input graph.*

5 Parameter $\eta + k$

We now show that MINIGL-ED is FPT parameterized by the neighborhood diversity η of the graph and the deletion budget k combined.

Theorem 5.1. *MINIGL-ED can be solved in $O((\eta + k)^{2k})$, where η is the neighborhood diversity of the input graph.*

Proof sketch. Let (G, k, T) be an instance of MINIGL-ED and $V_d = \{V_1, V_2, \dots, V_\eta\}$ be a neighborhood partition of G with $\eta = |V_d|$. We branch on each pair of neighborhood partitions $V_i, V_j \in V_d$ (without excluding the pairs where $i = j$) and arbitrarily delete an edge $u_i u_j$ where $u_i \in V_i$ and $u_j \in V_j$. After the edge is deleted we recompute the neighborhood partition by creating a new partition V'_i for the endpoints of the deleted edge. \square

| | |
|-------------------|---|
| Input | : Graph G , integer k , target IGL T , and twin cover C of G |
| Parameter: | $\Gamma = C , k$ |
| Output | : “yes” if (G, k, T) is a Yes-instance for MINIGL-ED, and “no” otherwise |
| 1 | Guess which subset $C' \subseteq E(C)$ to delete; |
| 2 | Compute \mathcal{P}_c over $V(G) \setminus C$; |
| 3 | Guess the number of edges $k' \leq k$ to delete from E_s ; |
| 4 | Guess which edges to delete from E_s ; |
| 5 | $\forall P_i \in \mathcal{P}_l$ guess the number of edges $k'' \leq k - k'$ to delete from $E_i(P_i)$; |
| 6 | foreach $P_i \in \mathcal{P}_l$ do |
| 7 | Guess the number of edges $k^* \leq k''$ to delete from $E_{Z_k}(P_i)$; |
| 8 | Guess which edges to delete from $E_{Z_k}(P_i)$; |
| 9 | Guess which $k'' - k^*$ edges to delete from $E_l(P_i)$; |
| 10 | if $IGL(G) \leq T$ then return yes; |
| 11 | return no; |

Algorithm 1: MINIGL-ED parameterized $\Gamma + k$.

6 Parameterized Intractability

In this section we present a W[1]-hardness result for MINIGL-ED for parameter tree-width (tw). We provide a parameterized reduction from the EQUITABLE COLORING (EC) problem. In EC, given a graph G and an integer r , the question is, does there exist a partitioning $\mathcal{V} = (V_1, V_2, \dots, V_r)$ of $V(G)$ into r independent sets such that the numbers of vertices in any two independent sets V_i, V_j differ by at most one? It is known that EC is W[1]-hard for parameter tree-width combined with the number of partitions r (Fellows et al. 2011).

Theorem 6.1. *MINIGL-ED is W[1]-hard for parameter $tw + k_d$.*

Proof. Let (G, r) be an instance of EC where G has tree-width tw . Assume without loss of generality that $l = \frac{n}{r}$ is an integer, where $n = |V(G)|$. We construct an instance $(G', k = n(r - 1), T)$ of MINIGL-ED where G' is defined as follows;

- create a set W of n vertices that contains a vertex corresponding to each vertex in $V(G)$,
- create a set $\mathcal{G} = \{G_1, G_2, \dots, G_r\}$ of r graphs where each $G_i \in \mathcal{G}$ is a copy of G , denote $U_G = \bigcup_{i=1}^r V(G_i)$,
- create a set R containing r vertices corresponding to each color in $\{1, 2, \dots, r\}$ (or shortly, $[r]$).
- create a set Q_v of n^3 vertices corresponding to each vertex $v \in U_G$ and connect them to v , denote $Q = |Q_v|$,
- create a set H_w of n^6 vertices corresponding to each vertex $w \in W$ and connect them to w , denote $H = |H_w|$,
- connect each vertex $w \in W$ to its corresponding vertex w_i in each $G_i \in \mathcal{G}$,
- for all $i \in [r]$ connect each vertex in $G_i \in \mathcal{G}$ to r vertices in R that correspond to a color i in $[r]$.

Let $U_Q = \bigcup_{v \in U_G} Q_v$ and $U_H = \bigcup_{w \in W} H_w$. It can be shown that $tw(G') \leq O(r^2 \cdot tw(G))$. We set $k = n(r - 1)$

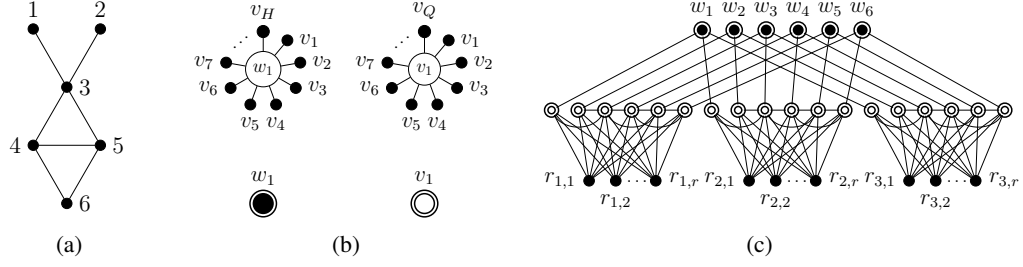


Figure 2: Depiction of (a) input graph G (b) vertices with H and Q leaves (c) graph G' obtained by transforming G .

and

$$\begin{aligned}
T &= \frac{1}{120} (10H^2 \cdot n(l-4) + 6H \cdot n(5n+4l+21) \\
&+ 10n \cdot r(4H+6Q(r+2)+3n+12r+3) + 20m(2l+3r) \\
&+ 5n(8n+3l+21) + 10l^2(3H+4) + 30r^2(r-1)). \quad (2)
\end{aligned}$$

Let us now show that (G, r) is Yes-instance if and only if (G', k, T) is a Yes-instance. Suppose, (G, r) is a Yes-instance and there exist a partition $\mathcal{V} = \{V_1, V_2, \dots, V_r\}$ such that for each $V_i \in \mathcal{V}$, $|V_i| = l$. We know that each $V_i \in \mathcal{V}$ is an independent set and for any pair of parts $V_i, V_j \in \mathcal{V}$, $V_i \cap V_j = \emptyset$. Therefore, we have a set of vertex covers $\{\tau_1, \tau_2, \dots, \tau_r\}$ where each $\tau_i = V(G) \setminus V_i$. Now, in each graph $G_i \in \mathcal{G}$, for every vertex v_i that correspond to a vertex $v \in \tau_i$, add the edge $v_i w$ to X , where $w \in W$. Observe that for each $v_i \in G_i$ $|N(v_i) \cap W| = 1$. Notice that $IGL(G' - X) = T$. Thus, (G', k, T) is a Yes-instance.

Conversely, suppose (G', k, T) is a Yes-instance and there exist a set $X \subseteq E(G)$ such that $IGL(G' - X) \leq T$. Let $\mathcal{E} = \{E_t, E_u, E_q, E_g, E_b\}$ be a partition of $E(G')$ where $E_u = (W \times U_G) \cap E(G')$, $E_g = E(G[U_G])$, $E_b = (R \times U_G) \cap E(G')$, $E_t = E(G[U_H \cup W])$, $E_q = E(G[U_Q \cup U_G])$. Denote $n' = |V(G')| = n(H+1) + nr(Q+1) + r^2$ and $\mathcal{K} = G' - (U_H \cup W)$ where \mathcal{K} consists of r isomorphic connected components K_1, K_2, \dots, K_r .

Claim 6.1.1. *Let (G', k, T) be an instance of MINIGL-ED. If X is an optimal solution, then $X \subseteq E_u$.*

Intuition. Observe that the endpoints of each edge in E_g and E_b are connected through many short alternative paths. Similarly, deleting an edge in E_t or E_q isolates a single vertex from G' . Comparatively, each edge in E_u has two high degree vertices as its endpoints that do not have a short alternative path. Also, deleting k edges from E_u can divide the graph in up to r connected components. \square

Now that we have established that we must choose $X \subseteq E_u$, we need to find the optimal set of edges in E_u . We know that $|E_u| = n \cdot r$ and $k' = n(r-1)$, this implies that we can delete all but n edges in E_u . Let E_u^w denotes the set of edges in E_u that are incident to $w \in W$. Observe that the edges in E_u^w connect w to its corresponding vertices in r copies of G . Thus for each $w \in W$, $|E_u^w| = r$.

Claim 6.1.2. *Let $X \subseteq E_u$ be an optimal solution for MINIGL-ED instance (as defined in Notions) on G' then $\forall w \in W$ in $G' - X$, $|E_u^w| = 1$.*

Intuition. We validate this by considering the following three cases for a vertex $w_i \in W$ in $G' - X$; (1) $|E_u^{w_i}| = 0$, (2) $|E_u^{w_i}| = 1$ and (3) $|E_u^{w_i}| > 1$. We know that $|W| = |E_u| - k = n$. Consequently, if we have a vertex $w_i \in W$ with $|E_u^{w_i}| = 0$, then there exist a vertex $w_j \in W$ with $|E_u^{w_j}| > 1$. Thus, case (1) implies case (3). Now, observe that each vertex $w_i \in W$ with $|E_u^{w_i}| > 1$ joins two components $K_i, K_j \in \mathcal{K}$. However, if for each vertex $w_i \in W$, $|E_u^{w_i}| = 1$, then the graph has r connected components. \square

Due to Claim 6.1.2, we have that $G' - X$ is divided into r connected components C_1, C_2, \dots, C_r , each containing a subgraph of the form $K_i \in \mathcal{K}$. Notice that if a pair of vertices $w_i, w_j \in W$ are in the same connected component C_i , their distance is at most 4. Thus, every distinct connected pair of vertices in $V(C_i) \cap W$, adds at least $\frac{H^2}{6} + \frac{2H}{5}$ to the IGL of $G' - X$. Therefore, we must choose X such that each component C_i in $G' - X$ contains the least number of vertices from W . Consequently, each connected component C_i contains a set $L_i \subseteq W$ of at most $l = \frac{n}{r}$ vertices.

Furthermore, inside each connected component C_i , the IGL of $G' - X$ can be reduced by maximizing the distance between the vertices in L_i . Observe that each vertex $w \in L_i$ is connected to its corresponding vertex in G_i (G_i is a copy of G in C_i) and each pair of vertices in G_i are at distance at most 2. Therefore, the distance between a pair of vertices in L_i can be maximized by choosing X such that the $N(L_i)$ is an independent set. But this is only possible if G has at least r disjoint independent sets of size l . Thus, (G, r) is a Yes-instance for EC. \square

7 Future Directions

Our results point to natural future directions: approximation and heuristic algorithms; a probabilistic version of the problem where each edge is associated by a probability, denoting the uncertainty with which it can be removed. Moreover, tractability for parameter k can also be considered. Notice that the NP and $W[1]$ hardness results obtained here will also apply to the probabilistic version.

Acknowledgements. Serge Gaspers is the recipient of an Australian Research Council (ARC) Future Fellowship (FT140100048) and acknowledges support under the ARC's Discovery Projects funding scheme (DP150101134). Kamran Najeebullah is supported by the Australian Government Research Training Program Scholarship.

References

- Aziz, H.; Gaspers, S.; Lee, E. J.; and Najeebullah, K. 2018. Defender Stackelberg game with inverse geodesic length as utility metric. In *Proc. of AAMAS*, 694–702.
- Aziz, H.; Gaspers, S.; and Najeebullah, K. 2017. Weakening covert networks by minimizing inverse geodesic length. In *Proc. of IJCAI*, 779–785.
- Carley, K. M.; Reminga, J.; and Kamneva, N. 2003. Destabilizing terrorist networks. In *Proc. of NAACSOS*. Wilcox Steven P.
- Chen, D.; Lü, L.; Shang, M.-S.; Zhang, Y.-C.; and Zhou, T. 2012. Identifying influential nodes in complex networks. *Physica A* 391(4):1777–1787.
- Chen, J.; Kanj, I. A.; and Xia, G. 2010. Improved upper bounds for vertex cover. *Theor. Comput. Sci.* 411(40-42):3736–3756.
- Diestel, R. 2010. *Graph Theory*. Graduate Texts in Math. Springer.
- Downey, R. G., and Fellows, M. R. 2013. *Fundamentals of Parameterized Complexity*. Springer.
- Drange, P. G.; Dregi, M. S.; and van’t Hof, P. 2014. On the computational complexity of vertex integrity and component order connectivity. In *Proc. of ISAAC*, 285–297.
- Fellows, M. R.; Lokshtanov, D.; Misra, N.; Rosamond, F. A.; and Saurabh, S. 2008. Graph layout problems parameterized by vertex cover. In *Proc. of ISAAC*, 294–305.
- Fellows, M. R.; Fomin, F. V.; Lokshtanov, D.; Rosamond, F. A.; Saurabh, S.; Szeider, S.; and Thomassen, C. 2011. On the complexity of some colorful problems parameterized by treewidth. *Inf. Comput.* 209(2):143–153.
- Fomin, F. V.; Liedloff, M.; Montealegre-Barba, P.; and Todinca, I. 2014. Algorithms parameterized by vertex cover and modular width, through potential maximal cliques. In *Proc. of SWAT*, 182–193.
- Ganian, R. 2012. Using neighborhood diversity to solve hard problems. Technical Report abs/1201.3091, arXiv CoRR.
- Ganian, R. 2015. Improving vertex cover as a graph parameter. *Discret. Math. Theor. C*. 17(2):77–100.
- Gaspers, S., and Lau, J. 2018. Minimizing and computing the inverse geodesic length on trees. Technical Report abs/1811.03836, arXiv CoRR.
- Gross, D.; Heinig, M.; Iswara, L.; Kazmierczak, L. W.; Luttrell, K.; Saccoman, J. T.; and Suffel, C. 2013. A survey of component order connectivity models of graph theoretic networks. *WSEAS Trans. Math.* 12:895–910.
- Holme, P.; Kim, B. J.; Yoon, C. N.; and Han, S. K. 2002. Attack vulnerability of complex networks. *Phys. Rev. E* 65(5). Art. no. 056109.
- Impagliazzo, R., and Paturi, R. 2001. On the complexity of k -sat. *J. Comput. Syst. Sci.* 62(2):367–375.
- Kovács, I. A., and Barabási, A.-L. 2015. Network science: Destruction perfected. *Nature* 524(7563):38–39.
- Kuhlman, C.; Tuli, G.; Swarup, S.; Marathe, M.; and Ravi, S. S. 2013. Blocking simple and complex contagion by edge removal. In *Proc. of ICDM*, 399–408.
- Lampis, M. 2012. Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica* 64(1):19–37.
- Latora, V., and Marchiori, M. 2001. Efficient behavior of small-world networks. *Phys. Rev. Lett.* 87(19):198701.
- Lokshtanov, D. 2015. Parameterized integer quadratic programming: Variables and coefficients. Technical Report abs/1511.00310, arXiv CoRR.
- Micali, S., and Vazirani, V. V. 1980. An $O(\sqrt{|V|} \cdot |E|)$ algorithm for finding maximum matching in general graphs. In *Proc. of FOCS*, 17–27.
- Michalak, T. P.; Rahwan, T.; Szczepanski, P. L.; Skibski, O.; Narayanam, R.; Jennings, N. R.; and Wooldridge, M. J. 2013. Computational analysis of connectivity games with applications to the investigation of terrorist networks. In *Proc. of IJCAI*, 293–301.
- Michalak, T. P.; Rahwan, T.; Skibski, O.; and Wooldridge, M. J. 2015. Defeating terrorist networks with game theory. *IEEE Intell. Syst.* 30(1):53–61.
- Michalak, T. P.; Rahwan, T.; and Wooldridge, M. J. 2017. Strategic social network analysis. In *Proc. of AAAI*.
- Morone, F., and Makse, H. A. 2015. Influence maximization in complex networks through optimal percolation. *Nature* 524:65–68.
- Sandeep, R. B., and Sivadasan, N. 2015. Parameterized lower bound and improved kernel for diamond-free edge deletion. In *Proc. of IPEC*, 365–376.
- Shamir, R.; Sharan, R.; and Tsur, D. 2004. Cluster graph modification problems. *Discrete Appl. Math.* 144(1-2):173–182.
- Szczepanski, P. L.; Michalak, T. P.; and Rahwan, T. 2015. Efficient algorithms for game-theoretic betweenness centrality. *Artif. Intell.* 231:39–63.
- UCINET Software. 2017. Datasets: Covert networks. Retrieved from <https://sites.google.com/site/ucinetsoftware/datasets/covert-networks> on 2017-02-17.
- Valente, T. W. 2012. Network interventions. *Science* 337(6090):49–53.
- Watts, D. J., and Strogatz, S. H. 1998. Collective dynamics of ‘small-world’ networks. *Nature* 393(6684):440–442.
- Zhang, H.; Jiang, Z.-H.; Wang, H.; Xie, F.; and Chen, C. 2012. Role of edges in complex network epidemiology. *Int. J. Mod. Phys. C* 23(09):1250059.
- Zhang, Y.; Adiga, A.; Saha, S.; Vullikanti, A.; and Prakash, B. A. 2016. Near-optimal algorithms for controlling propagation at group scale on networks. *IEEE Trans. Knowl. Data Eng.* 28(12):3339–3352.
- Zheng, A. X.; Dunagan, J.; and Kapoor, A. 2011. Active graph reachability reduction for network security and software engineering. In *Proc. of IJCAI*.