arXiv:2101.08042v1 [math.CO] 20 Jan 2021

The geodesic-transversal problem

Paul Manuel[a] Boštjan Brešar[b,c] Sandi Klavžar[b,c,d]

[a] Department of Information Science, College of Computing Science and Engineering, Kuwait University, Kuwait
pauldmanuel@gmail.com

[b] Faculty of Natural Sciences and Mathematics, University of Maribor, Slovenia
bostjan.bresar@um.si

[c] Institute of Mathematics, Physics and Mechanics, Ljubljana, Slovenia

[d] Faculty of Mathematics and Physics, University of Ljubljana, Slovenia
sandi.klavzar@fmf.uni-lj.si

Abstract

A maximal geodesic in a graph is a geodesic (alias shortest path) which is not a subpath of a longer geodesic. The geodesic-transversal problem in a graph G is introduced as the task to find a smallest set S of vertices of G such that each maximal geodesic has at least one vertex in S. The minimum cardinality of such a set is the geodesic-transversal number $\mathrm{gt}(G)$ of G. It is proved that $\mathrm{gt}(G) = 1$ if and only if G is a subdivided star and that the geodesic-transversal problem is NP-complete. Fast algorithms to determine the geodesic-transversal number of trees and of spread cactus graphs are designed, respectively.

Keywords: hitting set; geodesic-transversal problem; network centrality; tree; cactus graph; algorithm

AMS Subj. Class.: 05C69; 05C85; 68R10

1 Introduction

Given a set U and a family $S = \{S_1, \ldots, S_k\}$, where $S_i \subseteq U$, a subset H of U is a *hitting set* for the family S if $H \cap S_i \neq \emptyset$ for all $i \in \{1, \ldots, k\}$. The hitting set problem is to find a smallest hitting set for S. The hitting set problem is NP-complete [16] and has been studied in different terminologies. In particular, in graph theory the term S*-transversal problem* presents the quest for a minimum set of vertices that intersect every set of a given family S of subsets of the vertex set. When S is a collection of maximal cliques of a graph, the S-transversal problem is called the *clique-transversal problem* [1, 8, 9, 11, 12, 13, 18], and when S is a collection of fixed size cliques, it is called the *generalized clique transversal problem* [11, 12]. The clique-transversal problem is polynomially solvable for interval graphs and NP-complete for chordal graphs [12]. Dahlhaus et al. [13] have studied the

S-transversal problem where S is a collection of hyperedges in a hypergraph. When S is a collection of k-paths, the S-transversal problem is called the *k-path-transversal problem*. This problem has been well-studied under different terminologies [5, 6, 15, 19, 28, 30, 39, 47].

A *geodesic* in a graph G is a shortest path between two vertices, and a geodesic is *maximal* if it is not a subpath of a longer geodesic. When S is a collection of **maximal geodesics**, we call the S-transversal problem the *geodesic-transversal problem*. A geodesic on k vertices is a *k-geodesic*. When S is a collection of k-geodesics, the S-transversal problem is called *k-geodesic-transversal problem*.

To our knowledge, there is no literature on the geodesic-transversal problem and the k-geodesic-transversal problem. The *geodesic-transversal number* of G, denoted by $\mathrm{gt}(G)$, is the minimum cardinality of a geodesic-transversal set of G. A set S of vertices is a *gt-set* of G if S is a minimum cardinality geodesic-transversal set of G. Thus, the geodesic-transversal problem of G is to find a gt-set of G. It is easy to see that the 2-geodesic-transversal problem is the vertex cover problem.

In the next section, we provide further motivation for the new geodesic-transversal problem. In Section 3, we determine the geodesic-transversal number of some graphs and show that this number equals 1 precisely for subdivided stars. We also prove that the geodesic-transversal problem is NP-complete for general graphs. In Section 4 we derive a polynomial algorithm for arbitrary trees, while in Section 5 a fast algorithm is designed for spread cactus graphs.

2 Motivation from (large-scale) network theory

The geodesic-transversal problem is not entirely new. The path version of this problem is quite popular in graph theory and is well studied by graph theory researchers [5, 6, 15, 19, 28, 30, 39, 47]. A set S of vertices of a graph G is a *k-path vertex cover* if every path of order k in G contains at least one vertex from S [6]. It is not uncommon in graph theory that the same concept is studied under different names. If indeed so, this indicates that the concept is of wider interest. The k-path vertex cover has been studied also as vertex k-path cover [5], k-path vertex cover [2, 6, 19, 28, 30], VCP_k-set [39], and k-path cover [15]. The *k-path vertex cover problem* is to find the minimum cardinality of a k-path vertex cover. The problem is NP-hard for cubic planar graphs of girth 3 [6, 39] and for bipartite graphs [47]. The problem has applications in many areas, such as traffic control [41] and wireless sensor networks [6]. Funke et al. [15] have provided a list of applications of this problem on different domains. The concepts of path transversal have also been generalized to the context of hypergraphs [50]. The geodesic-transversal problem is a natural extension and adaptation of the path-transversal problem. Note that the k-path vertex cover problem and the k-geodesic transversal problem coincide in general graphs when $k = 2$, and coincide in triangle-free graphs when $k = 3$.

Betweenness centrality and closeness centrality are key measures of large-scale network analysis [32, 43]. The concepts of betweenness centrality and closeness centrality play a vital role in the study of large-scale network analysis including social networks [17, 26, 32], brain networks [14, 21, 25], biological networks (gene regulatory networks, protein-protein interaction network) [24, 25], chemical networks [49], communication networks [10], transport networks [27, 35] and IoT networks [35, 43] etc. The betweeness centrality $B(v)$ and closeness centrality $C(v)$ are defined as follows [32, 43]:

$$B(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

$$C(v) = \sum_{s \neq v \neq t} \sigma_{st}(v)$$

where σ_{st} is the total number of geodesics from node s to node t and $\sigma_{st}(v)$ is the number of those paths that pass-through v.

The scope of geodesic-transversal is wider than betweenness centrality and closeness centrality. The geodesic load geo-load(v) of a vertex v of a graph G is defined as the number of maximal geodesics which traverse through v. The concept of geo-load of a network is applied in the geodesics-based routing algorithms [34, 37]. The concept is also used in load-balanced routing of fixed interconnection networks [36, 46]. While the betweenness centrality of a vertex focuses on all possible geodesics, the geodesic load of a vertex concentrates on only maximal geodesics.

Some interesting combinatorial problems of large-scale network analysis are propagation (malware propagation [48], immunization [33], disease propagation [42] and data communication [20]), broadcasting, and gossiping problems [45]. An interesting research problem is to demonstrate how the geodesic-transversal is a good model to represent these problems in large-scale network analysis.

3 Basic observations and NP-completeness

For a starting example consider the Petersen graph P. It is of diameter 2, therefore to hit all the five maximal geodesics on the outer 5-cycle we need at least two vertices. Similarly, we need at least two vertices to hit the maximal geodesics which are subpaths of the inner 5-cycle. Hence $\mathrm{gt}(P) \geq 4$. On the other hand, in Fig. 1 a geodesic-transversal set with four vertices is shown, hence we conclude that $\mathrm{gt}(P) = 4$. Using a similar reasoning we can deduce that if $r, s \geq 1$, then $\mathrm{gt}(K_{r,s}) = \min\{r, s\}$.

The following simple lemma will turn out to be quite useful.

Lemma 3.1 *Let Q be a geodesic of a graph G and $x, y \in V(Q)$. If u is a vertex from $V(G) \setminus V(Q)$ such that $d(u,x) = d(u,y)$, then Q does not extend to a geodesic that contains u.*

Proof. Suppose on the contrary that Q' is a geodesic such that Q is contained in Q' and $u \in V(Q')$. Clearly, on the geodesic Q', the vertex u cannot lie between x and y. Therefore, either $d(u,x) < d(u,y)$ or $d(u,y) < d(u,x)$, and each of the possibilities in a contradiction with the lemma assumption. □

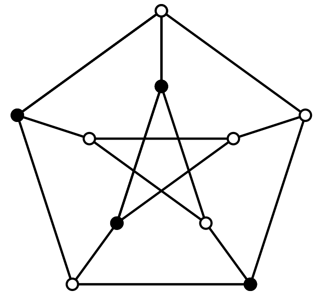

Figure 1: A gt-set of the Petersen graph

Clearly, $\mathrm{gt}(P_n) = 1$ for all $n \in \mathbb{N}$. In particular, $\mathrm{gt}(P_n) = 1$ because its only vertex forms a geodesic by itself and hence has to lie in its unique gt-set. Considering an arbitrary edge e of the complete graph K_n, $n \geq 3$, and a vertex not on the edge, Lemma 3.1 implies that at least one of the endpoint of e must lie in a geodesic-transversal set of K_n. Consequently, $\mathrm{gt}(K_n) = n - 1$ holds for $n \geq 2$. These two examples generalize as follows, where by a *subdivided star* we mean the graph obtained from $K_{1,k}$, $k \geq 1$, by subdividing each of the edges of $K_{1,k}$ arbitrary number of times (possibly zero). If $k = 1$, then the subdivided stars coincide with the family of paths.

Proposition 3.2 *If G is a connected graph of order at least 2, then $1 \leq \mathrm{gt}(G) \leq n(G) - 1$. In addition, the lower bound is attained if and only if G is a subdivided star, and the upper bound is attained if and only if G is a complete graph of order at least 2.*

Proof. Since every graph G has at least one maximal geodesic, we infer $\mathrm{gt}(G) \geq 1$. Since every maximal geodesic of a non-trivial graph contains at least two vertices, we infer $\mathrm{gt}(G) \leq n(G) - 1$.

Suppose now that $\mathrm{gt}(G) = 1$ and let $\{u\}$ be a gt-set of G. Let T be a BFS-tree of G with the root u.

We first claim that G is bipartite. Suppose on the contrary that there exists an edge xy of G, where vertices x and y lie in the k^{th} distance level of T, for some $k \geq 1$. Then $d_G(u,x) = d_G(u,y) = k$. Consider now an arbitrary maximal geodesic Q of G that contains the edge xy. Then Lemma 3.1 implies, that u does not belong to Q, a contradiction with the assumption that u forms a gt-set. Hence the claim.

We next claim that G is a tree. Suppose on the contrary that G contains at least one cycle C. Since we already know that G is bipartite, considering the cycle C we infer that there exist a vertex x of C which lies in some k^{th} distance level of T such that x has two neighbors (in G), say y and z, in the $(k-1)^{\mathrm{st}}$ distance level of T. If Q is an arbitrary maximal geodesic of G that contains as a subpath the path $y - x - z$, then Lemma 3.1 again implies, that u does not belong to Q, a contradiction. Hence G is a tree.

We finally claim that G is a subdivided star. If this is not the case, then in T (which is just G, rooted in u), there exists a vertex x which lies in k^{th} distance level of T, $k \geq 1$, such that x has two neighbors, say y and z, in the $(k+1)^{\mathrm{st}}$ distance level of T. As in the previous paragraph we now see that a maximal geodesic of G that contains as a subpath the path $y-x-z$, yields a contradiction. It follows that every vertex of T, except maybe u, is of degree either 2 or 1. The latter is is equivalent to the fact that T is a subdivided star. We hence conclude that $\mathrm{gt}(G)=1$ holds if and only if G is a subdivided star.

Suppose now that G is a an arbitrary graph that is not complete. Then there exist vertices $x, y \in V(G)$ such that $xy \notin E(G)$. But then $V(G) \setminus \{x, y\}$ form a geodesic-transversal set of G and consequently, $\mathrm{gt}(G) \leq n(G)-2$. We can hence conclude that $\mathrm{gt}(G)=n(G)-1$ and and only if G is a complete graph of order at least 2. □

To conclude the section we are going to show that the geodesic-transversal problem is NP-complete. In the study of vertex-deletion problems [47], the concept of a *dissociation set* (see [4, 22, 40]) was considered, which was shown in [6] to be the complement of a 3-path vertex cover in any graph. Since dissociation set problem is NP-complete even when restricted to bipartite graphs [47], we infer the following.

Theorem 3.3 [6, 47] *The* 3*-path vertex cover problem is NP-complete for bipartite graphs.*

For additional complexity results on the 3-path vertex cover problem see [3, 23, 38, 44]. It is clear that in bipartite graphs the 3-path vertex cover and the 3-geodesic transversal coincide. Thus, Theorem 3.3 can be restated as follows:

Theorem 3.4 *The* 3*-geodesic-transversal problem is NP-complete for bipartite graphs.*

Now we will prove that the geodesic-transversal problem is NP-complete for general graphs. In order to prove this, we will provide a polynomial reduction from the 3-geodesic-transversal problem to the geodesic-transversal problem. Given a graph G, where $V(G) = [n] = \{1, \ldots, n\}$, the reduced graph is denoted by G', where $V(G') = V \cup \{x, y, z\}$ and $E(G') = E \cup \{xz, zy\} \cup \{iz : i \in V\}$. For an example see Fig. 2.

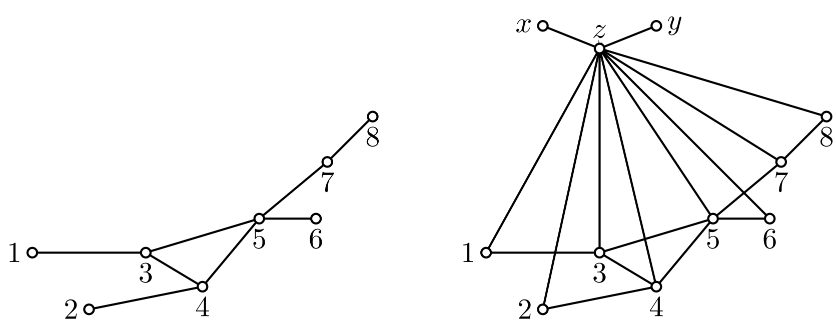

Figure 2: A graph (left) and its reduced graph (right)

Property 3.5 *A set S of vertices is a* 3*-geodesic-transversal of G if and only if $S \cup \{z\}$ is a geodesic-transversal of G'.*

Property 3.5 leads to the following conclusion:

Theorem 3.6 *The geodesic-transversal problem is NP-complete for general graphs.*

4 The geodesic-transversal problem of trees

In this section, we design an algorithm to locate a gt-set of a tree.

Let T be a tree. A vertex of degree 1 of a tree is a *leaf*. A neighbor of a leaf is a *support vertex*. A support vertex u is an *end support vertex* if u is adjacent to at least $\deg(u) - 1$ leafs.

Lemma 4.1 *A tree of order at least* 2 *has at least one end support vertex.*

Proof. Let T be a tree of order at least 2 and let $u_1, \dots u_k$ be the support vertices of T. Let T' be a tree obtained from T by removing all the leaves of T. Suppose that $\deg_{T'}(u_i) \geq 2$ for for each $i \in [k]$. Since the degree of every vertex of $T' \setminus \{u_1, \dots, u_k\}$ is the same in T' as in T, we would have a tree T' whose every vertex is of degree at least 2. As this is clearly not possible, there exists a vertex u_i such that $\deg_{T'}(u_i) \leq 1$. This in turn means that u_i is an end support vertex of T. □

Let G be a graph, let $v \in V(G)$ be a vertex of degree 2, and let x and y be the neighbors of u. If G' is the graph obtained from G be removing the vertex u and adding the edge xy, then we say that G' is obtained from G by *smoothing* the vertex u. Note that if the vertices u, x, and y induce a triangle in G, then there are two parallel edges between x and y in G'. Let further $\mathrm{SM}(G)$ denote a graph obtained from G by smoothing all the vertices of G of degree 2. Since the smoothing operation preserves the degree of vertices, $\mathrm{SM}(G)$ is well-defined, that is, unique up to isomorphism. In particular, no matter in which order a smoothing of vertices of C_n, $n \geq 3$, is performed, we end up with $\mathrm{SM}(C_n) = C_2$. (The 2-cycle C_2 is the graph on the vertices with two parallel edges.) For another example see Fig. 3.

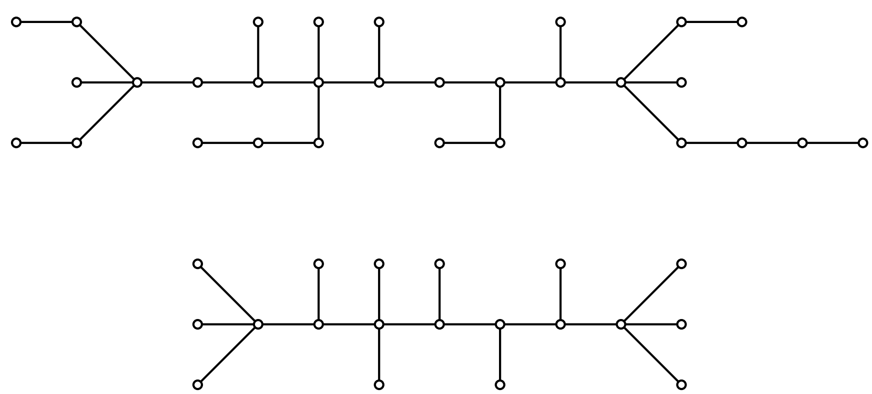

Figure 3: A tree T (above) and $\mathrm{SM}(T)$ (below)

Lemma 4.2 *If T is a tree, then* $\mathrm{gt}(T) = \mathrm{gt}(\mathrm{SM}(T))$.

Proof. Let S be a gt-set of T. Suppose that S contains a vertex u with $\deg(u) = 2$. Let P be the maximal path of T that contains u and exactly two vertices which are not of degree 2. Such a path is indeed unique. To see it, let x and y be the neighbors of u. If $\deg(x) = 2$, then continue the path until the first vertex which is not of degree 2 is found. Such a vertex exists since T is a tree. Do the same procedure from the vertex y. Now, every maximal geodesic in T that contains u, also contains x and y. It follows that $(S \setminus \{u\}) \cup \{x\}$ (or $(S \setminus \{u\}) \cup \{y\}$ for that matter) is also a gt-set of T. Repeating this construction for every vertex of S of degree 2 we arrive at a gt-set S' of T which contains no vertex of degree 2. Since $S' \subseteq V(\mathrm{SM}(T))$ is also a gt-set of $\mathrm{SM}(T)$, it follows that $\mathrm{gt}(\mathrm{SM}(T)) \leq \mathrm{gt}(T)$. On the other hand, if S is a gt-set of $\mathrm{SM}(T)$, then we infer that S is also a gt-set of T, hence $\mathrm{gt}(T) \leq \mathrm{gt}(\mathrm{SM}(T))$ also holds. □

Lemma 4.2 does not hold for an arbitrary graph G, even when $SM(G)$ does not contain parallel edges. See Fig. 4, where a graph G is show for which we have $\mathrm{gt}(G) = 4$ and $\mathrm{gt}(\mathrm{SM}(G)) = 3$.

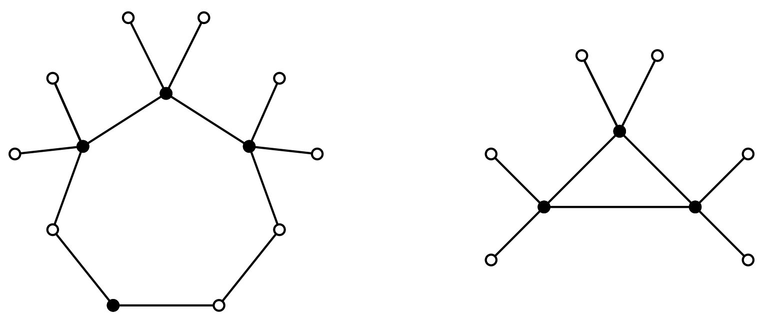

Figure 4: A graph G (left) with $\mathrm{gt}(G) = 4$, and $\mathrm{SM}(G)$ (right) with $\mathrm{gt}(\mathrm{SM}(G)) = 3$

Lemma 4.3 *Let T be a tree with no vertices of degree 2. Let u be an end support vertex of T and $u_1, \ldots, u_s$ the leaves adjacent to u. Then* $\mathrm{gt}(T) = \mathrm{gt}(T \setminus \{u, u_1, \ldots, u_s\}) + 1$. *Moreover, there exists a gt-set S of T such that $u \in S$.*

Proof. Since T has no vertices of degree 2, the end support vertex u is adjacent to at least two leaves, that is, $s \geq 2$. If T is a star, and hence u being the center of it, then the assertion of the lemma is clear. In the rest of the proof we may thus assume that u has at least one non-leaf neighbor, and since u is an end support vertex, it has only one non-leaf neighbor. We denote the latter vertex by w, and let T' be the component of $T - u$ that contains the vertex w.

Let S be a gt-set of T. Since $s \geq 2$, we see that $|S \cap \{u, u_1, \ldots, u_s\}| \geq 1$, for otherwise the geodesic u_1, u, u_2 would not be hit. Moreover, $|S \cap \{u, u_1, \ldots, u_s\}| = 1$. If $u_i \in S$ for some $i \in [s]$, then $(S \setminus \{u_i\}) \cup \{u\}$ is also a gt-set of T. This proves the last assertion of the lemma and we may without loss of generality assume in the rest that $u \in S$.

We claim now that $S \cap V(T')$ is a gt-set of T'. Indeed, since $\deg_{T'}(w) \geq 2$, no maximal geodesic of T' can be hit by u. That is, only the vertices from T' can be used to hit the maximal geodesics of T', hence the claim. It follows that $\mathrm{gt}(T) = 1 + \mathrm{gt}(T') = 1 + \mathrm{gt}(T \setminus \{u, u_1, \ldots, u_s\})$ and we are done. □

Here, an algorithm is designed to construct a gt-set S of an arbitrary tree T.

Algorithm 1: A gt-set of a tree

Input: A tree T.
Output: A gt-set S of T.

```
1 S = ∅;
2 T = SM(T) (i.e., perform the smoothing operation on each vertex of degree 2 in
  T).
3 while |V(T)| > 0 do
4     identify an arbitrary end support vertex p of SM(T);
5     S = S ∪ {p};
6     T = T \ {p, p_1 ..., p_t}, where p_1, ..., p_t are leaf neighbors of p;
7     T = SM(T).
```

Theorem 4.4 *Given a tree T, Algorithm 1 determines a gt-set of T in linear time.*

The proof of correctness of Algorithm 1 follows from Lemmas 4.1, 4.2, and 4.3. The time complexity of the algorithm is clearly linear.

To see that the smoothing operation performed in Line 2 and Line 7 of Algorithm 1 is necessary, consider the tree T in Fig. 5. Note first that $\mathrm{SM}(T) = 4$. Assuming that Line 2 and Line 7 would be removed from the algorithm, the modified algorithm would return a wrong value 5. On the other hand, Algorithm 1 first produces $\mathrm{SM}(T)$. Then, after two while loops (after selecting two end support vertices), another smoothing operation at Line 7 is needed. This in turn guarantees that the algorithm will end after two additional selections of end support vertices, and hence will return the correct value 4.

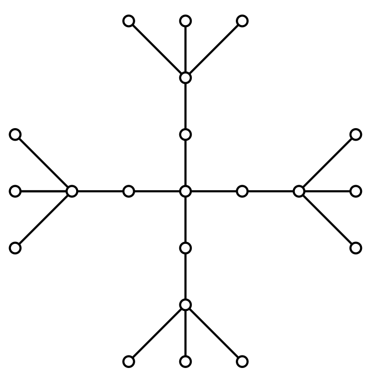

Figure 5: Tree T

5 Fast algorithm on spread cactus graphs

A connected graph in which each edge belongs to at most one cycle is a *a cactus graph*. We further restrict our attention to the subclass of cactus graphs in which every vertex belongs to at most one cycle, and call them *spread cactus graphs*. They are exactly the graphs that have neither a diamond nor a butterfly as a topological minor [31]. Every block in these graphs is either K_2 or a cycle, and cycle blocks do not intersect other cycle blocks. The blocks in a spread cactus have a tree structure, and they contain leaves or *leaf-cycles*, where the latter are defined as the cycle blocks, which intersect only one K_2-block.

As usual, let C_n denote an n-cycle. Let C be an n-cycle with vertices $\{v_1, \ldots, v_n\}$, and let $I \subseteq [n]$ be a set of indices of vertices in $V(C)$. By $C_n(I)$ we denote the graph obtained from C by attaching a leaf v_i' to the vertex $v_i \in V(C)$ for every $i \in I$. If $I = \{i_1, \ldots, i_k\}$, then we will simplify the notation $C_n(\{i_1, \ldots, i_k\})$ to $C_n(i_1, \ldots, i_k)$. For instance, $C_3(1, 2, 3)$ denotes the net graph, $C_3(1, 2)$ is known as the bull graph, $C_3(1)$ is the paw graph, while $C_4(1)$ is the P-graph; see Fig. 6 for the former three graphs.

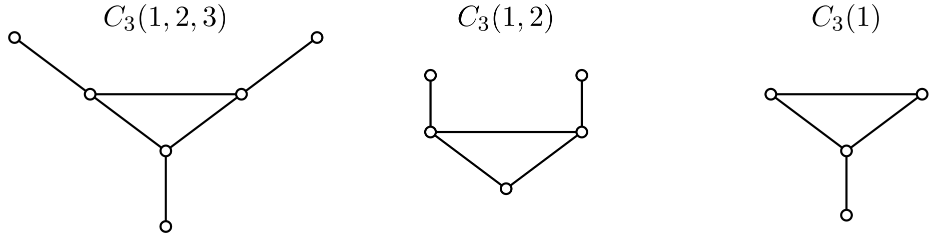

Figure 6: Net, bull, and paw

We start our discussion by constructing an algorithm that finds a minimum geodesic transversal in the graphs $C_n(I)$ for all $n \geq 3$ and any index set $I \subseteq [n]$. Note that $C_n(I)$ are spread cactus graphs with only one cycle and no two K_2-blocks intersect.

Consider $C_n(I)$, where $I = \{i_1, \ldots, i_k\}$ and $i_1 < i_2 < \cdots < i_k$. In the following, these indices will be taken modulo k. If $j \in [k]$, then we set P^j to be a $v_{i_j}, v_{i_{j+1}}$-path along $C_n(I)$, that is, the path on vertices $v_{i_j}, v_{i_j+1}, \ldots, v_{i_{j+1}}$. If $j = k$, this thus means that P^k is the path on vertices $v_{i_k}, v_{i_k+1}, \ldots, v_1, \ldots, v_{i_1}$.

We claim that there exists a gt-set S of $C_n(I)$ such that each path P^j, $j \in [k]$, contains a vertex in S. Indeed, if

$$i_{j+1} - i_j \leq \left\lfloor \frac{n}{2} \right\rfloor,$$

then P^j lies on the maximal geodesic between v_j' and v_{j+1}'. Now, if a gt-set S contains v_j' (resp., v_{j+1}'), then $S' = (S - \{v_j'\}) \cup \{v_j\}$ (resp., $S' = (S - \{v_{j+1}'\}) \cup \{v_{j+1}\}$) is clearly a gt-set of $C_n(I)$. On the other hand, if

$$i_{j+1} - i_j > \left\lfloor \frac{n}{2} \right\rfloor,$$

then either P^j contains a maximal geodesic between two vertices in C, or there is a maximal geodesic between v'_j and v_{j+1}. Hence we may assume that P^j contains a vertex in S.

To state the next lemma, we introduce the following concept. In the graph $C_n(I)$, where $I = \{i_1, \ldots, i_k\}$, we say that $j \in [k]$ is *lonely*, if $i_{j+1} - i_{j-1} > \lfloor \frac{n}{2} \rfloor + 1$.

Lemma 5.1 *If $n \geq 3$ and $I = \{i_1, \ldots, i_k\}$, where $0 \leq k \leq n$, then*

$$\mathrm{gt}(C_n(I)) = \begin{cases} 2; & k \leq 3, \\ \frac{k+1}{2}; & k \geq 5 \text{ odd}, \\ \frac{k}{2} + 1; & k \geq 4 \text{ even, and there exist lonely } j_1, j_2 \in [k], j_1 \text{ odd}, j_2 \text{ even}, \\ \frac{k}{2}; & \text{otherwise}. \end{cases}$$

Proof. Set $G = C_n(I)$ and use the notation for vertices of G as established before the lemma. Let S be a gt-set of G. Then, as noted above, we may assume that $S \cap V(G) \subseteq C$.

We start with the case $k = |I| = 0$, that is, $G = C_n$. In this case, $S = \{v_1, v_i\}$, where $i = \lfloor \frac{n}{2} \rfloor$, is clearly a gt-set of G, yielding $\mathrm{gt}(G) = 2$. When $k \in \{1, 2\}$, and assuming without loss of generality that $1 \in I$, again the set $S = \{v_1, v_i\}$, where $i = \lfloor \frac{n}{2} \rfloor$, is a gt-set of G. Next, let $k = 3$, and assume without loss of generality that $1 \in I$. If the set $S = \{v_1, v_i\}$, where $i = \lfloor \frac{n}{2} \rfloor$, is not a gt-set of G, then we may assume that $1 < i_2 < i_3 < \lfloor \frac{n}{2} \rfloor$ (the case when $\lfloor \frac{n}{2} \rfloor < i_2 < i_3$ can be dealt with in a similar way). However, then $S = \{v_2, v_{2 + \lfloor \frac{n}{2} \rfloor}\}$ is a gt-set of G, yielding $\mathrm{gt}(G) = 2$. The first line of the equality of the lemma is thus established. We next consider $k \geq 4$ and distinguish two cases.

Let k be odd, $k \geq 5$. Assume that for every even $j \in [k]$, we have $i_{j+1} - i_{j-1} \leq \lfloor \frac{n}{2} \rfloor + 1$. Then the set $S = \{v_{i_j} : i_j \in I \text{ and } j \text{ odd}\}$ is a gt-set of G with $|S| = \frac{k+1}{2}$. Indeed, since a maximal geodesic in C_n is of length $\lfloor \frac{n}{2} \rfloor$, every maximal geodesic in $C_n(I)$ has at least one leaf as an endvertex, from which we derive that it contains a vertex v_{i_j}, where j is odd. In the second case we may assume without loss of generality that $i_3 - i_1 > \lfloor \frac{n}{2} \rfloor + 1$. Then $S = \{v_i : i = i_3 - \lfloor \frac{n}{2} \rfloor - 1 \text{ or } i > 1 \text{ odd}\}$ is a gt-set of G with $|S| = \frac{k+1}{2}$.

Finally, let k be even, $k \geq 4$. Suppose first that for every even $j \in [k]$ we have $i_{j+1} - i_{j-1} \leq \lfloor \frac{n}{2} \rfloor + 1$. Then, we derive in the same way as in the case of odd k that the set $S = \{v_{i_j} : i_j \in I \text{ and } j \text{ odd}\}$ is a gt-set of G with $|S| = \frac{k}{2}$. In a similar way we conclude that $\mathrm{gt}(G) = \frac{k}{2}$ if for every odd $j \in [k]$ we have $i_{j+1} - i_{j-1} \leq \lfloor \frac{n}{2} \rfloor + 1$. In the second case there exist a lonely odd $j_1 \in [k]$ and a lonely even $j_2 \in [k]$. Then the path P^t between $v_{i_{j_t - 1}}$ and $v_{i_{j_t + 1}}$ is of length at least $\lfloor \frac{n}{2} \rfloor + 2$, which implies that this path contains a maximal geodesic of length $\lfloor \frac{n}{2} \rfloor$, which does not involve $v_{i_{j_t+1}}$ nor $v_{i_{j_t-1}}$. Since a gt-set must hit both paths P^t, we infer that $\mathrm{gt}(G) > \frac{k}{2}$. It is easy to see that $\mathrm{gt}(G) \leq \frac{k}{2} + 1$ by using a similar construction as in the case when k is odd. □

From the proof it is also clear that a gt-set of a graph $C_n(I)$ can be efficiently computed. If the set I is a part of the input, the computation can be done in time linear in the size of I.

Next, we determine a minimum geodesic transversal set S in a graph $C_n(I)$ in which some of the vertices are declared in advance to be in S. This situation appears naturally in the construction of an algorithm for determining a gt-set of a unicyclic graph presented later.

Let $A \subseteq [n]$ be the set of indices of the vertices of the cycle of $C_n(I)$ such that every v_i, $i \in A$, is predetermined to be in a geodesic transversal set S of $C_n(I)$. Denote by $C_n(I, A)$ the graph $C_n(I)$ together with the requirement that vertices indexed by elements from A must lie in a geodesic transversal set. The algorithm for constructing a minimum geodesic transversal of $C_n(I, A)$ is based on the constructions from the proof of Lemma 5.1. In Algorithm 2, the notation of vertices $v_i \in V(C_n)$ is simplified to i. The indices from $A = \{a_1, \ldots, a_t\}$ are ordered cyclically as follows:

$$a_1 < a_2 < \cdots < a_t < a_{t+1} = a_1,$$

by which the main `while` loop is performed at least once (and is performed exactly once when $A = \{a_1\}$). The correctness of Algorithm 2 can be proved by using similar arguments as in the proof of Lemma 5.1.

We continue by presenting an algorithm for determining a gt-set of a unicyclic graph. (This part is written mostly for intuition purposes. Algorithm 3 deals also with the special case when G is unicyclic.) Let G be a unicyclic graph, and C the cycle in G of length n. If G is isomorphic to C_n, then $\mathrm{gt}(G) = 2$. Otherwise, let $G' = G - E(C)$, let $T_1, \ldots, T_r$ be the nontrivial components of G', and let $v_1, \ldots, v_r$ be the vertices of C, where v_i belongs to T_i for all $i \in [r]$. Clearly, each T_i is a tree on at least two vertices. If T_i is a path, then by the smoothing operation, and the fact that $\mathrm{gt}(\mathrm{SM}(T_i)) = \mathrm{gt}(T_i)$, we may assume that T_i is isomorphic to P_2, that is, v_i has a leaf attached. In this case we set $S_i = \emptyset$. Otherwise, T_i has vertices of degree at least 3, and we perform the algorithm for obtaining a gt-set S_i of a tree T_i. It is easy to see that the sets S_i, $i \in [r]$, are subsets of a gt-set of G. There are three possibilities:

(i) $v_i \in S_i$;

(ii) $v_i \notin S_i$, but all neighbors of v_i in T_i are in S_i;

(iii) $v_i \notin S_i$, and there is a neighbor of v_i in T_i that is not in S_i.

Turning back our attention to G, after gt-sets of trees T_i are obtained, the above possibilities yield different cases by which we complete the construction of a gt-set of G. Note that all maximal geodesics within trees T_i are hit by the sets S_i, hence it remains to consider the maximal geodesics that pass some vertices of C. The problem can be translated to determination of a minimum geodesic transversal of $C_n(I, A)$. In particular, all vertices v_i that are in S_i (possibility (i)) are considered to be in the set A, all vertices v_i that are not in S_i and have a neighbor in T_i that is not in S_i (possibility (iii)) are considered to be in I. Finally, the vertices $v_i \notin S_i$ for which possibility (ii) appears are in neither of the sets A and I (the same holds for the vertices of C that are isolated in G'). Perform Algorithm 2 on $C_n(I, A)$, and let S be the output of the algorithm. Finally, $S' = S \cup \bigcup_{i=1}^{r} S_i$ is a gt-set of G.

Algorithm 2: A minimum geodesic transversal of $C_n(I, A)$

Input: Cycle on $V(C_n) = \{1, \ldots, n\}$, a leaf attached to i, where $i \in I$, and $A \subseteq [n]$.

Output: Minimum geodesic transversal S of $C_n(I)$ containing A.

$S = A$;
Order $A : a_1 < a_2 < \cdots < a_t < a_{t+1} = a_1$;
$i = 1$;
while $i \leq t$ **do**
    let $I_i = \{j \in I :\ a_i < j < a_{i+1}\} = \{j_1, \ldots, j_k\}$ and $j_0 = a_i, j_{k+1} = a_{i+1}$;
    if k *odd* **then**
        if $\forall \ell \in [\frac{k+1}{2}] :\ j_{2\ell} - j_{2(\ell-1)} \leq \lfloor \frac{n}{2} \rfloor + 1$ **then**
            $S = S \cup \{j_{2\ell} :\ \ell \in [\frac{k-1}{2}]\}$;
        else
            let $m \in [\frac{k+1}{2}]$, where $j_{2m} - j_{2m-2} > \lfloor \frac{n}{2} \rfloor + 1$;
            $S = S \cup \{j_{2\ell} :\ \ell \in [\frac{k-1}{2}]\} \bigcup \{j_{2m-2} + \lfloor \frac{n}{2} \rfloor + 1\}$;
    else
        let $\ell = 0$;
        while $\ell \leq k$ **do**
            if $j_{\ell+2} - j_\ell \leq \lfloor \frac{n}{2} \rfloor + 1$ **then**
                $S = S \bigcup \{j_{\ell+2}\}; \ell = \ell + 2$;
            else
                $S = S \bigcup \{j_\ell + \lfloor \frac{n}{2} \rfloor + 1\}$;
                if $j_{\ell+1} - j_\ell \leq \lfloor \frac{n}{2} \rfloor + 1$ **then**
                    $S = S \bigcup \{j_{\ell+3}\}; \ell = \ell + 3$;
                else
                    $S = S \bigcup \{j_{\ell+2}\}; \ell = \ell + 2$;
    $i = i + 1$;

We follow with two auxiliary results that will be a key for the algorithm for determining a gt-set of a spread cactus graph. We need some more notation. A vertex v in a graph G is *heavy* if $\deg_G(v) \geq 3$. Next, a heavy vertex v is a *boundary heavy vertex* if at most one component of $G - v$ is not a path. If v is a heavy vertex, then let P^v denote the subset of $V(G)$ containing v and every vertex of degree at most 2 that can be reached from v on a path that does not contain heavy vertices.

Lemma 5.2 *If G is a graph and v a boundary heavy vertex in G such that $G-v$ has more than two components, then* $\mathrm{gt}(G) = 1 + \mathrm{gt}(G - P^v)$.

Proof. Since P_t contains two leaves, there is a maximal geodesic that lies in P_v. Hence

$\mathrm{gt}(G) \geq 1 + \mathrm{gt}(G - P^v)$. Since every maximal geodesic in G that contains a vertex in P^v contains also v, we infer $\mathrm{gt}(G) = 1 + \mathrm{gt}(G - P^v)$. $\square$

Consider now a graph G in which some of the vertices are declared to be in a geodesic transversal, and denote by A_G the set of such vertices in G. (This situation appears naturally within an algorithm for determining a gt-set of G, where in the process of building a gt-set some of the vertices are already put in the set.) Let $C : v_1, \ldots, v_n, v_1$ be a cycle in G, let $A = A_G \cap V(C)$, and let I be the set of vertices v_i, $i \in [n]$, which are adjacent to a leaf. We say that C is a *boundary cycle* in G if there exists at most one vertex $v_j \in V(C)$, where $v_j \notin I \cup A$, such that v_i has a neighbor outside C.

Lemma 5.3 *Let G be a graph, C a boundary cycle in G, I support vertices of C, A the set of vertices in C that belong to A_G, and $x \in V(C)$ be adjacent to a non-leaf vertex outside C. Let S_C be a minimum geodesic transversal of $C_n(I, A)$ and S'_C a minimum geodesic transversal of $C_n(I \cup \{x\}, A)$. If $|S'_C| = |S_C|$, then S'_C belongs to a minimum geodesic transversal of G that contains A_G. Otherwise, $|S'_C| = |S_C| + 1$, and S_C belongs to a minimum geodesic transversal of G that contains A_G.*

Proof. Clearly, $|S_C| \leq |S'_C| \leq |S_C| + 1$. A (minimum) geodesic transversal of G must hit all maximal geodesics between two vertices in $C_n(I)$. This implies that at least $|S_C|$ vertices from C need to be in a minimum geodesic transversal of G that contains A_G. If $|S'_C| = |S_C|$, then S'_C is a better choice than S_C, since it hits not only all the maximal geodesics that lie between two vertices in $C_n(I)$, but also all maximal geodesics that have one endvertex in $C_n(I)$. Otherwise, when $|S'_C| = |S_C| + 1$, S_C belongs to a minimum geodesic transversal of G that contains A_G. $\square$

A gt-set of a path clearly consist of a single vertex, hence we may concentrate on spread cactus graphs that are not paths. Note that for such graphs there exists a boundary heavy vertex or a boundary cycle. Hence, using Lemmas 5.2 and 5.3, we propose Algorithm 3 for determining a gt-set of a spread cactus graph.

Theorem 5.4 *Given a spread cactus graph G, which is not a path, Algorithm 3 determines a gt-set of G in linear time.*

Proof. By the above observations, if G is a non-path spread cactus graph, then G contains a heavy vertex v. Now, there are three possibilities: v is a boundary heavy vertex that does not lie on a cycle (Line 3), v lies on a cycle and its degree is at least 4 (Line 18), or v lies on a cycle and its degree is 3. (By Line 17, v can be made adjacent to a leaf.) If the latter holds for all heavy vertices of a cycle with at most one exception, then we have a boundary cycle (Line 5). The correctness of the first case and the second case (Line 3 and 18, resp.) follows from Lemma 5.2, the correctness of the second case (Lines 5-13) follows from Lemma 5.3. The case when v is a boundary heavy vertex with degree 3 that lies on a cycle (Lines 16-17) follows similar arguments as in the proof of Lemma 4.2.

An implementation of the algorithm uses a tree-like structure of a spread cactus graph, which can be obtained by a BFS search. Finding a boundary heavy vertex can be done

Algorithm 3: A minimum geodesic transversal of a spread cactus graph G.

Input: A spread cactus graph G, which is not a path.

Output: Minimum geodesic transversal S of G.

1 $S = \emptyset$;
2 **while** *there is a heavy vertex in* G **do**
3   **if** *there is a boundary heavy vertex* v *that lies on no cycle* **then**
4     $S = S \cup \{v\}$; $G = G - P^v$;
5   **else if** *there is a boundary cycle* $C = C_n(I, A)$, *where* $A = V(C) \cap S$, **then**
6   **if** x *a vertex in* C *with a non-leaf neighbor* **then**
7   let S_C a minimum geodesic transversal of $C_n(I, A)$ and S'_C a minimum geodesic transversal of $C_n(I \cup \{x\}, A)$;
8   **if** $|S_C| = |S'_C|$ **then**
9         $S = S \cup S'_C$; remove from G all vertices of $C_n(I)$ and all vertices of degree 2 reachable by a path from x;
10   **else**
11         $S = S \cup S_C$; $G = G - \Big(V(C_n(I)) \setminus \{x\}\Big)$;
12   **else**
13       $G = C_n(I, A)$, where $A = V(G) \cap S$, and let S' be a minimum geodesic transversal of G containing A; $S = S \cup S'$;
14   **else**
15   let v be a boundary heavy vertex lying on a cycle;
16   **if** $\deg(v) = 3$ **then**
17   smooth out the path P_v so that v is adjacent to a leaf
18   **else**
19       $S = S \cup \{v\}$;
20       $G = G - P^v$.

by using a reversed order of the BFS, and all cases of the `if-then-else` condition can be checked in linear time with respect to the number of vertices that they involve. In particular, the case when there is a boundary cycle (lines 5-13) can be realized in linear time by applying Algorithm 2 twice. $\square$

6 Conclusion and future work

A new concept of geodesic-transversal is introduced in this paper. In addition to NP-completeness, polynomial time algorithms are derived for arbitrary trees and spread cactus graphs. The potential future research is to investigate the complexity status of this problem for important interconnection networks such as butterfly networks and hypercubes, as well as for other classes of graphs such as bipartite graphs and chordal graphs. As mentioned in the initial part of the paper, it would be interesting to study how the geodesic-transversal can be used to model distance-based combinatorial problems in large-scale network analysis.

Acknowledgments

This work was supported and funded by Kuwait University, Research Project No. (QI 01/20).

References

[1] T. Andreae, M. Schughart, Zs. Tuza, Clique-transversal sets of line graphs and complements of line graphs, Discrete Math. 88 (1991) 11–20.

[2] C. Brause, R. Krivoš-Belluš, On a relation between k-path partition and k-path vertex cover, Discrete Appl. Math. 223 (2017) 28–38.

[3] C. Brause, I. Schiermeyer, Kernelization of the 3-path vertex cover problem, Discrete Math. 339 (2016) 1935–1939.

[4] B. Brešar, B. L. Hartnell, D. F. Rall, Uniformly dissociated graphs, Ars Math. Contemp. 13 (2017) 293–306

[5] B. Brešar, M. Jakovac, J. Katrenič, G. Semanišin, A. Taranenko, On the vertex k-path cover, Discrete Appl. Math. 161 (2013) 1943–1949.

[6] B. Brešar, F. Kardoš, J. Katrenič, G. Semanišin, Minimum k-path vertex cover, Discrete Appl. Math. 159 (2011) 1189–1195.

[7] B. Brešar, R. Krivoš-Belluš, G. Semanišin, P. Šparl, On the weighted k-path vertex cover problem, Discrete Appl. Math. 177 (2014) 14–18.

[8] V. Balachandhran, P. Nagavamsi, C. Pandu Rangan, Clique transversal and clique independence on comparability graphs, Inform. Process. Lett. 58 (1996) 181–184.

[9] M.-S. Chang, Y.-H. Chen, G.J. Chang, J.-H. Yan, Algorithmic aspects of the generalized clique-transversal problem on chordal graphs, Discrete Appl. Math. 66 (1996) 189–203.

[10] V. Chellappan, K.M. Sivalingam, Application of entropy of centrality measures to routing in tactical wireless networks, 2013 19th IEEE Workshop on Local & Metropolitan Area Networks (LANMAN), Brussels, 2013, 1–6.

[11] D.G. Corneil, J. Fonlupt, The complexity of generalized clique covering, Discrete Appl. Math. 22 (1988/89) 109–118.

[12] E. Dahlhaus, J. Kratochvíl, P.D. Manuel, M. Miller, Parallel algorithms for generalized clique transversal problems, Australas. J. Combin. 33 (1995) 3–14.

[13] E. Dahlhaus, J. Kratochvíl, P.D. Manuel, M. Miller, Transversal partitioning in balanced hypergraphs, Discrete Appl. Math. 79 (1997) 75–90.

[14] A. Fornito, A. Zalesky, E.T. Bullmore, Fundamentals of Brain Network Analysis, Academic Press, 2016.

[15] S. Funke, A. Nusser, S. Storandt, On k-path covers and their applications, Proceedings of the VLDB Endowment 7 (2014) 893–902.

[16] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, New York, 1979.

[17] D. Gómez, J.R. Figueira, A. Eusébio, Modeling centrality measures in social network analysis using bi-criteria network flow optimization problems, European J. Oper. Res. 226 (2013) 354–365.

[18] V. Guruswami,C.P. Rangan, Algorithmic aspects of clique-transversal and clique-independent sets, Discrete Appl. Math. 100 (2000) 183–202.

[19] D.A. Hoang, A. Suzuki, T. Yagita, Reconfiguring k-path vertex covers, Lecture Notes Comp. Sci. 12049 (2020) 133–145.

[20] M. Hofmann, T. Braun, G. Carle, Multicast communication in large scale networks, in: Third IEEE Workshop on the Architecture and Implementation of High Performance Communication Subsystems, Mystic, CT, USA, 1995, 147–150.

[21] K.E. Joyce, P.J. Laurienti, J.H. Burdette, S. Hayasaka, A new measure of centrality for brain networks, PLoS ONE 5 (2010) e12200.

[22] F. Kardoš, J. Katrenič, I. Schiermeyer, On computing the minimum 3-path vertex cover and dissociation number of graphs, Theoret. Comput. Sci. 412 (2011) 7009–7017.

[23] J. Katrenič, A faster FPT algorithm for 3-path vertex cover, Inform. Process. Lett. 116 (2016) 273–278.

[24] D. Koschützki, F. Schreiber, Centrality analysis methods for biological networks and their application to gene regulatory networks, Gene Regul. Syst. Bio. 2 (2008) 193–201.

[25] H. Kwon, Y-H. Choi, J.-M. Lee, A physarum centrality measure of the human brain network, Sci. Reports 9 (2019) Article 5907.

[26] C. Li, Q. Li, P. Van Mieghem, H.E. Stanley, H. Wang, Correlation between centrality metrics and their application to the opinion model, European Phys. J. B 88 (2015) Article 65.

[27] W. Liu, X. Li, T. Liu, B. Liu, Approximating betweenness centrality to identify key nodes in a weighted urban complex transportation network, J. Adv. Transp. (2019) Article 9024745.

[28] P. Manuel, Revisiting path-type covering and partitioning problems, arXiv:1807.10613 [math.CO] (25 July 2018).

[29] P. Manuel, On the isometric path partition problem, Discussiones Mathematicae Graph Theory, pp 1-13, 2019, DOI: https://doi.org/10.7151/dmgt.2236.

[30] D.B. Mokeev, D.S. Malyshev, A polynomial-time algorithm of finding a minimum k-path vertex cover and a maximum k-path packing in some graphs, Opt. Lett. 14 (2020) 1317–1322.

[31] E.A. Nordhaus, R.D. Ringeisen, B.M. Stewart, A.T. White, A Kuratowski-type theorem for the maximum genus of a graph, J. Combin. Theory Ser. B 12 (1972) 260–267.

[32] C. Perez, R. Germon, Chapter 7 - Graph Creation and Analysis for Linking Actors: Application to Social Data, in: R. Layton, P.A. Watters (eds.), Automating Open Source Intelligence, Syngress, 2016, 103–129.

[33] B.A. Prakash, Propagation and immunization in large networks, ACM Mag. Stud. 19 (2012) 56–59.

[34] H. Racke, Minimizing congestion in general networks, in: The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002, Vancouver, BC, 2002, 43–52.

[35] Y. Ren, Betweenness Centrality and Its Applications from Modeling Traffic Flows to Network Community Detection, PhD Thesis, University of Notre Dame, USA, 2015.

[36] A. Singh, Load-balanced routing in interconnection networks, PhD thesis, Stanford University, USA, 2005.

[37] M. Stewart, R. Kannan, A. Dvir, B. Krishnamachari, CASPaR: Congestion avoidance shortest path routing for delay tolerant networks, Int. J. Dist. Sensor Networks 13(11) (2017) 1–15.

[38] D. Tsur, Parameterized algorithm for 3-path vertex cover, Theoret. Comput. Sci. 783 (2019) 1–8.

[39] J. Tu, F. Yang, The vertex cover P_3 problem in cubic graphs, Inform. Process. Lett. 113 (2013) 481–485.

[40] J. Tu, Z. Zhang Y. Shi, The maximum number of maximum dissociation sets in trees, J. Graph Theory (2020) doi.org/10.1002/jgt.22627.

[41] J Tu, W. Zhou, A primal–dual approximation algorithm for the vertex cover P_3 problem, Theoret. Comput. Sci. 412 (2011) 7044–7048.

[42] X. Wang, W. Ni, K. Zheng, R.P. Liu, X. Niu, Virus propagation modeling and convergence analysis in large-scale networks, in: IEEE Trans. Inform. Forensics Sec. 11 (2016) 2241–2254.

[43] W. Xi, J. Han, K. Li, Z. Jiang, H. Ding, Chapter 13 - Location Inferring in Internet of Things and Big Data, in: R. Buyya, R.N. Calheiros, A.V. Dastjerdi (eds.), Big Data, Morgan Kaufmann, 2016, 309–335.

[44] M. Xiao, S. Kou, Exact algorithms for the maximum dissociation set and minimum 3-path vertex cover problems, Theoret. Comput. Sci. 657 (2017) 86–97.

[45] J.-M. Xu, Topological Structure and Analysis of Interconnection Networks, Springer, 2002.

[46] Z. Xu, X. Huang, Y. Deng, Load-balanced routing for nested interconnection networks, arXiv:1909.06497v2 [cs.NI] (25 Dec 2019).

[47] M. Yannakakis, Node-deletion problems on bipartite graphs, SIAM J. Comput. 10 (1981) 310–327.

[48] S. Yu, G. Gu, A. Barnawi, S. Guo, I. Stojmenovic, Malware propagation in large-scale networks, IEEE Trans. Know. Data Eng. 27 (2015) 170–179.

[49] P. Zhao, S.M. Nackman, C.K. Law, On the application of betweenness centrality in chemical network analysis: Computational diagnostics and model reduction, Combust. Flame 162 (2015) 2991–2998.

[50] A. Żak, Generalized transversals, generalized vertex covers and node-fault-tolerance in graphs, Discrete Appl. Math. 255 (2019) 299–306.