On the Terminal Connection Problem

Alexsander A. de Melo[1(✉)], Celina M. H. de Figueiredo[1], and Uéverton S. Souza[2]

[1] Federal University of Rio de Janeiro, Rio de Janeiro, Brazil
{aamelo,celina}@cos.ufrj.br
[2] Fluminense Federal University, Niterói, Brazil
ueverton@ic.uff.br

Abstract. A *connection tree* of a graph G for a *terminal set* W is a tree subgraph T of G such that leaves$(T) \subseteq W \subseteq V(T)$. A non-terminal vertex of a connection tree T is called *linker* if its degree in T is exactly 2, and it is called *router* if its degree in T is at least 3. The Terminal connection problem (TCP) asks whether G admits a connection tree for W with at most ℓ linkers and at most r routers, while the Steiner tree problem asks whether G admits a connection tree for W with at most k non-terminal vertices. We prove that TCP is NP-complete even when restricted to strongly chordal graphs and $r \geq 0$ is fixed. This result separates the complexity of TCP from the complexity of Steiner tree, which is known to be polynomial-time solvable on strongly chordal graphs. In contrast, when restricted to cographs, we prove that TCP is polynomial-time solvable, agreeing with the complexity of Steiner tree. Finally, we prove that TCP remains NP-complete on graphs of maximum degree 3 even if either $\ell \geq 0$ or $r \geq 0$ is fixed.

Keywords: Terminal vertices · Connection tree · Steiner tree · Strongly chordal graphs · Cographs · Bounded degree

1 Introduction

Steiner tree is one of the most fundamental network design problems, proved to be NP-complete by Karp in his seminal paper [17]. Besides being related to several real-world applications, Steiner tree is of great theoretical interest, and it has been extensively studied from the perspective of graph theory [4,8,13,25,29] and computational complexity [2,7,11,26]. Formally, the Steiner tree problem has as input a connected graph G, a non-empty terminal set $W \subseteq V(G)$, and a non-negative integer k, and it asks whether there exists a tree subgraph T of G such that $W \subseteq V(T)$ and $|V(T) \setminus W| \leq k$. In this paper, we analyse the computational complexity of a network design problem closely related to Steiner tree, called Terminal connection.

Let G be a graph and $W \subseteq V(G)$ be a non-empty set. A *connection tree* T of G for W is a tree subgraph of G such that leaves$(T) \subseteq W \subseteq V(T)$. In a connection tree T for W, the vertices belonging W are called *terminal*, and the

T. Bureš et al. (Eds.): SOFSEM 2021, LNCS 12607, pp. 278–292, 2021.
https://doi.org/10.1007/978-3-030-67731-2_20

vertices belonging to $V(T) \setminus W$ are called *non-terminal* and are classified into two types according to their respective degrees in T, namely: the non-terminal vertices of degree exactly equal to 2 in T are called *linkers* and the non-terminal vertices of degree at least 3 in T are called *routers* cf. [9]. We remark that the vertex set of every connection tree can be partitioned into terminal vertices, linkers and routers. For each connection tree T, we let $\mathsf{L}(T)$ denote the linker set of T and $\mathsf{R}(T)$ denote the router set of T. Next, we present a formal definition for the Terminal connection problem.

Terminal Connection (TCP)

Input:	A connected graph G, a non-empty terminal set $W \subseteq V(G)$ and two non-negative integers ℓ and r.
Question:	Does there exist a connection tree T of G for W such that $\lvert\mathsf{L}(T)\rvert \leq \ell$ and $\lvert\mathsf{R}(T)\rvert \leq r$?

TCP was introduced by Dourado et al. [9], having as motivation applications in information security and network routing, and it was proved to be polynomial-time solvable when the parameters ℓ and r are both fixed [9]. Nevertheless, it was proved to be NP-complete even if either $\ell \geq 0$ or $r \geq 0$ is fixed [9]. In particular, the problem was proved to be NP-complete even if $\ell \geq 0$ is fixed and the input graph has constant maximum degree [10].

There is a straightforward Turing reduction from Steiner tree to TCP, namely: (G, W, k) is a yes-instance of Steiner tree if and only if (G, W, ℓ, r) is a yes-instance of TCP for some pair $\ell, r \in \{0, \ldots, k\}$ such that $\ell + r = k$. An interesting aspect of this Turing reduction is the fact that it preserves the structure of the input graph. Consequently, if TCP is polynomial-time solvable on some graph class $\mathcal{G}$, then so is Steiner tree. Analogously, if Steiner tree is NP-complete on some graph class $\mathcal{G}$, then TCP cannot be solved in polynomial-time on $\mathcal{G}$, unless P=NP. Nevertheless, possibly Steiner tree is polynomial-time solvable on some graph class $\mathcal{G}$ whereas TCP remains NP-complete on $\mathcal{G}$.

In this work, we confirm the existence of such a non-trivial graph class. More specifically, we prove that TCP remains NP-complete on *strongly chordal* graphs, while it is known that Steiner tree is polynomial-time solvable on *strongly chordal* graphs [29]. On the other hand, we prove that TCP can be solved in polynomial-time on *cographs*, agreeing with the computational complexity of Steiner tree [4]. Finally, we prove that TCP remains NP-complete on planar graphs of maximum degree 3 even if either $\ell \geq 0$ or $r \geq 0$ is fixed. It is worth mentioning that TCP can be trivially solved in polynomial-time when restricted to graphs of maximum degree 2. Thus, our result establishes an *NP-complete versus polynomial-time dichotomy* for the problem with respect to the maximum degree of the input graph. Moreover, we note that, although it is known that Steiner tree is NP-complete on planar graphs of maximum degree 3 [19], our result cannot be seen as an immediate consequence of such a fact. Indeed, possibly TCP is polynomial-time solvable on a graph class $\mathcal{G}$ if either the parameter $\ell \geq 0$ or the parameter $r \geq 0$ is fixed, while Steiner tree remains NP-complete on $\mathcal{G}$.

Related Works. Motivated by applications in optical networks and bandwidth consumption minimization, another variant of Steiner tree that has been investigated is the one in which the number of *branching nodes* in the sought tree T, i.e. vertices (which not necessarily are non-terminal) of degree at least 3 in T, is bounded. In [14,27,28], the authors addressed the undirected and directed cases of this variant, for which they devised approximation and parameterized tractable algorithms, apart from obtaining some intractability results.

In addition, Dourado et al. introduced in [10] the *strict* variant of TCP, so-called S-TCP, which has the same input of TCP but further requires that the sought connection tree T satisfies leaves$(T) = W \subseteq V(T)$. It is worth mentioning that, just as TCP can be seen as a generalization of Steiner tree, S-TCP can be seen as a generalization of Full Steiner tree, which is a widely studied variant of Steiner tree [15,18,20]. Similarly to TCP, it was proved that S-TCP is polynomial-time solvable when the parameters $\ell \geq 0$ and $r \geq 0$ are both fixed [10], and that the problem is still NP-complete if $\ell \geq 0$ is fixed [10]. Nevertheless, except for the case $r \in \{0, 1\}$, which was shown to be polynomial-time solvable [22], the complexity of S-TCP for fixed $r \geq 2$ has remained open. Motivated by this question, S-TCP was also investigated in [21,23]. In particular, in [23], S-TCP was proved to be NP-complete (and W[2]-hard when parameterized by r), even if $\ell \geq 0$ is constant and the input graph is restricted to split graphs. An interesting fact of this proof is that it can be easily adapted to TCP. Consequently, we obtain that TCP is also NP-complete (and W[2]-hard when parameterized by r) on split graphs. Besides this result, it was analysed in [23] the complexity of S-TCP when restricted to graphs of bounded maximum degree, and it was also proved that S-TCP is polynomial-time solvable on cographs.

Graph Notation. For any missing definition or terminology, we refer to [3]. In this work, all graphs are finite, simple and undirected. Let G be a graph. We let $V(G)$ and $E(G)$ denote the vertex set and the edge set of G, respectively. For every vertex $u \in V(G)$, we let $N_G(u)$ and $N_G[u] = N_G(u) \cup \{u\}$ denote the *(open) neighbourhood* and the *closed neighbourhood* of u in G, respectively; and we let $d_G(u) = |N_G(u)|$ denote the *degree* of u in G. Two distinct vertices $u, v \in V(G)$ are said to be *true twins* in G if $N_G[u] = N_G[v]$. A vertex $u \in V(G)$ is called a *universal vertex* of G if $N_G[u] = V(G)$. The *length* of a path P is defined as the number of edges of P. The *distance* between two vertices $u, v \in V(G)$ is the length of a path of G between u and v of minimum length. For every non-empty subset $S \subseteq V(G)$, we let $G[S]$ denote the *subgraph of G induced by S*.

Due to space restrictions, throughout this work, proofs of statements marked with $(\star)$ are omitted.

2 Strongly Chordal Graphs

In this section, we prove that, for each $r \geq 0$, TCP remains NP-complete when restricted to strongly chordal graphs.

A *chord* of a cycle C is an edge between any two non-consecutive vertices of C. A graph G is called *chordal* if every cycle of G of length at least 4 has a chord.

In other words, a graph G is chordal if every induced cycle of G has length 3. An *even cycle* is a cycle of even length. A chord uv of an even cycle C is called an *odd chord* if the distance between u and v in C is odd. A graph G is called *strongly chordal* if it is chordal and every even cycle of G of length at least 6 has an odd chord. A vertex u is called a *simple vertex* of a graph G if, for any two vertices $v, v' \in N_G(u)$, $N_G[v] \subseteq N_G[v']$ or $N_G[v'] \subseteq N_G[v]$. In other words, a vertex u of a graph G is simple if the collection $\{N_G[v] \mid v \in N_G(u)\}$ can be linearly ordered by set inclusion. Farber [12] proved that a graph G is strongly chordal if and only if there exists a linear order $\langle u_1, \dots, u_n \rangle$ of the vertices of G, called *simple elimination ordering*, such that u_i is a simple vertex of $G[\{u_i, \dots, u_n\}]$ for each $i \in \{1, \dots, n\}$, where n denotes the number of vertices of G.

In order to prove that TCP remains NP-complete on strongly chordal graph, we present a polynomial-time reduction from the Hamiltonian path problem on strongly chordal graphs, which was shown to be NP-complete by Müller [24]. Actually, we prove in Proposition 1 that Hamiltonian path problem can be reduced in polynomial-time to st-Hamiltonian path on strongly chordal graphs, and then we present in Theorem 1 (built on Construction 2) a polynomial-time reduction from st-Hamiltonian path to TCP on strongly chordal graphs.

The Hamiltonian path problem has as input a graph G and asks whether G admits a *Hamiltonian path*, i.e. a path that contains all vertices of G; and the st-Hamiltonian path problem is the variant of Hamiltonian path which has as input a graph G and two distinct vertices s and t and asks whether G admits a st-Hamiltonian path, i.e. a Hamiltonian path between s and t.

Lemma 1. *The class of strongly chordal graphs is closed under the operation of adding universal vertices.*

Proof. Let G be a strongly chordal graph and let $\langle u_1, \dots, u_n \rangle$ be a simple elimination ordering of G. Also, let H be the graph obtained from G by adding a universal vertex v. One can verify that $\langle u_1, \dots, u_n, v \rangle$ is a simple elimination ordering of H. Therefore, H is strongly chordal. □

Proposition 1. st-Hamiltonian path *remains* NP-*complete when restricted to strongly chordal graphs in which s and t have degree* 1 *each.*

Proof. Let G be a strongly chordal graph and let G' be the graph obtained from G by adding two universal vertices v and v', adding two new vertices s and t, and by adding the edges sv and $v't$. Based on Lemma 1, it is not hard to check that G' is strongly chordal. Furthermore, by construction, s and t have degree 1 in G' each. Finally, we note that $\langle u_1, \dots, u_n \rangle$ is a Hamiltonian path of G if and only if $\langle s, v, u_1, \dots, u_n, v', t \rangle$ is a st-Hamiltonian path of G'. □

Construction 1. (Gadget H_r and Terminal Set W_r). Let r be a positive integer. We define the gadget H_r as the graph such that

$$V(H_r) = \{\rho_1, \dots, \rho_r\} \cup \{x_1^1, x_1^2\} \cup \{x_i \mid i \in \{2, \dots, r\}\} \text{ and}$$
$$E(H_r) = \{\rho_i \rho_{i+1} \mid i \in \{1, \dots, r-1\}\} \cup \{x_1^1 \rho_1, x_1^2 \rho_1\} \cup \{x_i \rho_i \mid i \in \{2, \dots, r\}\}.$$

Moreover, we let $W_r = \{x_1^1, x_1^2\} \cup \{x_2, \dots, x_r\}$ be the terminal set of H_r.

Construction 2. (Reduction from st-HAMILTONIAN PATH to TCP). Let G be a graph and $s, t \in V(G)$ be two distinct vertices of G. Based on Proposition 1, assume without loss of generality that $d_G(s) = d_G(t) = 1$. Additionally, assume that $V(G) = \{u_1, \dots, u_n\}$, for some positive integer n, where $u_1 = s$ and $u_n = t$. Let r be a non-negative integer. We let G' be the graph obtained from G, s, t and r as follows:

– Add all vertices and all edges of G to G';
– Add new vertices s' and t' and add the edges $s's$ and tt';
– For each vertex $u_i \in V(G) \setminus \{s, t\}$, add a true twin u_i' of u_i, in such a way that $N_{G'}[u_i'] = N_{G'}[u_i]$;
– For each vertex $u_i \in V(G) \setminus \{s, t\}$, add a new vertex w_i and add the edges $u_i w_i$ and $u_i' w_i$, where u_i' denotes the true twin of u_i added in the last step;
– If $r \geq 1$, create the gadget H_r and the terminal set W_r described in Construction 1, and add the edge $\rho_r s'$; if $r = 0$, define $W_r = \emptyset$.

Then, we let $g(G, s, t, r) = (G', W, \ell, r)$ be the instance of TCP such that $W = \{s', t'\} \cup \{w_2, \dots, w_{n-1}\} \cup W_r$ and $\ell = 2n - 2$.

Lemma 2. *Let G be a graph and $s, t \in V(G)$ be two distinct vertices of G. Assume that s and t have degree 1 in G each. For each $r \geq 0$, G admits a st-Hamiltonian path if and only if the instance $g(G, s, t, r)$ described in Construction 2 is a* yes*-instance of TCP.*

Proof. Assume that $g(G, s, t, r) = (G', W, \ell, r)$. Moreover, for simplicity, consider $W_r = V(H_r) = E(H_r) = \emptyset$ if $r = 0$. First, suppose that there exists in G a st-Hamiltonian path $P = \langle s, u_{j_2}, \dots, u_{j_{n-1}}, t\rangle$. Then, let T be the graph with vertex set $V(T) = V(H_r) \cup V(P) \cup \{s', t'\} \cup \{w_{j_2}, u_{j_2}', \dots, w_{j_{n-1}}, u_{j_{n-1}}'\}$ and edge set

$$\begin{aligned} E(T) = E(H_r) &\cup \{\rho_r s'\} \cup \{s's\} \cup \{s u_{j_2}\} \\ &\cup \{u_{j_2} w_{j_2}, w_{j_2} u_{j_2}', \dots, u_{j_{n-1}} w_{j_{n-1}}, w_{j_{n-1}} u_{j_{n-1}}'\} \cup \{u_{j_{n-1}}' t\} \cup \{tt'\}, \end{aligned}$$

where u_{j_i}' denotes the true twin of u_{j_i} added in the construction of G'. Note that T is a connection tree of G' for W with $\mathsf{L}(T) = \{s, u_{j_2}, u_{j_2}', \dots, u_{j_{n-1}}, u_{j_{n-1}}', t\}$ and $\mathsf{R}(T) = \{\rho_1, \dots, \rho_r\}$. Therefore, $g(G, s, t, r)$ is a yes-instance of TCP.

Conversely, suppose that $g(G, s, t, r)$ is a yes-instance of TCP. Let T be a connection tree of G' for W such that $|\mathsf{L}(T)| \leq 2n - 2$ and $|\mathsf{R}(T)| \leq r$. We note that ρ_1 is the only neighbour of the terminal vertices $x_1^1, x_1^2 \in W_r$ and, for each $i \in \{2, \dots, r\}$, ρ_i is the only neighbour of the terminal vertex $x_i \in W_r$. As a result, T must contain all the vertices $\rho_1, \dots, \rho_r$. More specifically, such vertices must be routers of T. Consequently, $T' = T - H_r$ cannot contain any router, and all non-terminal vertices of T' must be linkers. Moreover, by construction, s' and t' have degree 1 in T'. This implies that the vertices $s, w_2, \dots, w_{n-1}, t$ have degree exactly 2 in T' each, otherwise T would not be connected or $W \not\subseteq V(T)$. Hence, T' consists in a path P' between s' and t' of the form

$$P' = \langle s', s, u_{j_2}, w_{j_2}, u_{j_2}', \dots, u_{j_{n-1}}, w_{j_{n-1}}, u_{j_{n-1}}', t, t'\rangle,$$

where u'_{j_i} denotes the true twin of u_{j_i} added in the construction of G'. Therefore, G admits a st-Hamiltonian path. Indeed, $\langle s, u_{j_2}, \ldots, u_{j_{n-1}}, t\rangle$ is a st-Hamiltonian path of G. $\square$

Lemma 3. *The class of strongly chordal graphs is closed under the operation of adding true twins.*

Proof. Let G be a strongly chordal graph and let $\langle u_1, \ldots, u_n\rangle$ be a simple elimination ordering of G. Also, let v be a vertex of G and H be the graph obtained from G by adding a true twin v' of v. Suppose that $v = u_i$, for some $i \in \{1, \ldots, n\}$. One can readily verify that $\langle u_1, \ldots, u_i, v', \ldots, u_n\rangle$ is a simple elimination ordering of H. Therefore, H is strongly chordal. $\square$

Lemma 4. *Let G be a strongly chordal graph with two true twin vertices v and v'. The graph G' obtained from G by adding a new vertex w and adding the edges vw and $v'w$ is strongly chordal.*

Proof. Let $\langle u_1, \ldots, u_n\rangle$ be a simple elimination ordering of G. Since by construction $N_{G'}(w) = \{v, v'\}$ and $N_G[v] = N_G[v']$, it is immediate that $\langle w, u_1, \ldots, u_n\rangle$ is a simple elimination ordering of G'. Therefore, G' is strongly chordal. $\square$

Theorem 1 ($\star$). *For each $r \geq 0$, TCP remains NP-complete when restricted to strongly chordal graphs.*

3 Cographs

In this section, we prove that TCP is linear-time solvable when restricted to cographs. A *cograph*, also called *complement reducible graph*, is a graph that does not contain a path of length 3 as an induced subgraph. Alternatively, cographs are characterized by the following recursive definition, given by Corneil et al. [5]:

– A graph on a single vertex is a cograph;
– If $G_1, \ldots, G_k$ are cographs, then so is their *disjoint union* $G_1 \cup \cdots \cup G_k$, i.e. the graph with vertex set $V(G_1 \cup \cdots \cup G_k) = V(G_1) \uplus \cdots \uplus V(G_k)$ and edge set $E(G_1 \cup \cdots \cup G_k) = E(G_1) \uplus \cdots \uplus E(G_k)$;
– If G is a cograph, then so is its complement $\overline{G}$.

We note that, if $\overline{G}$ is a cograph on more than one vertex, then there exist $k \geq 2$ cographs $G_1, \ldots, G_k$ such that G is their *join* $G_1 \wedge \cdots \wedge G_k$, i.e. the graph with vertex set $V(G_1 \wedge \cdots \wedge G_k) = V(G_1 \cup \cdots \cup G_k)$ and edge set $E(G_1 \wedge \cdots \wedge G_k) = E(G_1 \cup \cdots \cup G_k) \uplus \{uv \mid u \in V(G_i),\, v \in V(G_j),\, i, j \in \{1, \ldots, k\},\, i \neq j\}$.

An interesting property of cographs is the fact that every cograph G can be uniquely represented (up to isomorphism) by a rooted tree $\mathcal{T}_G$, called *cotree*, such that the leaves of $\mathcal{T}_G$ correspond to the vertices of G, and each internal node u of $\mathcal{T}_G$ corresponds to either the disjoint union or the join of the cographs induced by the leaves of the subtrees of $\mathcal{T}_G$ rooted at each child of u [5]. Another important property is that, given a graph G, recognising G as a cograph, as well as obtaining its respective cotree (if any), can be performed in time linear in the number of vertices and the number of edges of G [6].

Let $I = (G, W, \ell, r)$ be an instance of TCP, where G is a cograph. Since TCP can be easily solved in linear-time if $|W| < 3$ or $G[W]$ is connected, we assume throughout this section that $|W| \geq 3$ and $G[W]$ is not connected. Moreover, we assume that G is connected. Therefore, G must be the join of $k \geq 2$ cographs $G_1, \ldots, G_k$.

Lemma 5 ($\star$). *Let G be a cograph that is the join of $k \geq 2$ cographs $G_1, \ldots, G_k$, and let $W \subseteq V(G)$ be a terminal set such that $|W| \geq 3$ and $G[W]$ is not connected. Then, there exists a unique $i \in \{1, \ldots, k\}$ such that $V(G_i) \cap W \neq \emptyset$. Moreover, G admits a connection tree for W that contains exactly one router and no linker.*

Considering the input graph G as the join of $k \geq 2$ cographs $G_1, \ldots, G_k$, it follows from Lemma 5 that TCP can be trivially solved if $r \geq 1$, or $V(G_i) \cap W \neq \emptyset$ and $V(G_j) \cap W \neq \emptyset$ for some $i, j \in \{1, \ldots, k\}$, with $i \neq j$. Thus, we dedicate the remainder of this section to resolve the case in which $r = 0$ and there exists a unique $i \in \{1, \ldots, k\}$ such that $V(G_i) \cap W \neq \emptyset$.

Lemma 6. *Let G be a cograph and $W \subseteq V(G)$ be a non-empty terminal set. If T is a connection tree of G for W such that $\mathsf{R}(T) = \emptyset$ and $|\mathsf{L}(T)|$ is minimum, then $N_T(u) \subseteq W$ for each $u \in \mathsf{L}(T)$.*

Proof. Let T be a connection tree of G for W such that $\mathsf{R}(T) = \emptyset$ and $|\mathsf{L}(T)|$ is minimum. For the sake of contradiction, suppose that $N_T(u) \not\subseteq W$ for some linker $u \in \mathsf{L}(T)$. Since $\mathsf{R}(T) = \emptyset$ and leaves$(T) \subseteq W$, u belongs to a path P of T between two terminal vertices $w, w' \in W$, such that $(V(P) \setminus \{w, w'\}) \cap W = \emptyset$. Thus, it follows from the assumption $N_T(u) \not\subseteq W$ that $|V(P)| \geq 4$. Since cographs do not contain paths of length 3 as induced subgraphs, there exists a path P' of G between w and w' such that $|V(P')| \leq 3$ and $V(P') \subseteq V(P)$. Then, let T' be the graph with vertex set $V(T') = (V(T) \setminus V(P)) \cup V(P')$ and edge set $E(T') = (E(T) \setminus V(P)) \cup E(P')$. One can easily verify that T' is a connection tree of G for W such that $\mathsf{R}(T) = \emptyset$ and $\mathsf{L}(T') \subsetneq \mathsf{L}(T)$, which contradicts the minimality of $|\mathsf{L}(T)|$. $\square$

For each graph G, we let $\mathsf{cc}(G)$ denote the set of connected components of G, and we let $o(G) = |\mathsf{cc}(G)|$ denote the number of connected components of G.

Corollary 1 ($\star$). *Let G be a cograph, $W \subseteq V(G)$ be a non-empty terminal set, and let T be a connection tree of G for W such that $\mathsf{R}(T) = \emptyset$. If $|\mathsf{L}(T)|$ is minimum, then $|\mathsf{L}(T)| = o(G[W]) - 1$.*

Corollary 1 establishes that, whenever a cograph G admits a connection tree for a non-empty terminal set $W \subseteq V(G)$ that does not contain routers, G admits a connection tree T for W such that $\mathsf{R}(T) = \emptyset$ and $\mathsf{L}(T) = o(G[W]) - 1$. More importantly, it establishes that $o(G[W]) - 1$ is the minimum possible number of linkers that such a tree T can have. Therefore, if $I = (G, W, \ell, r)$ is an instance of TCP such that G is a cograph and $r = 0$, then ℓ must be at least $o(G[W]) - 1$, otherwise I is certainly a no-instance of the problem.

A *connection forest* of a graph G for a non-empty terminal set W is a subgraph F of G such that F is a forest and $\bigcup_{T \in \mathsf{cc}(F)} \text{leaves}(T) \subseteq W \subseteq V(F)$. A connection forest F is said to be *routerless* if $\bigcup_{T \in \mathsf{cc}(F)} \mathsf{R}(T) = \emptyset$. For each graph G and each non-empty terminal $W \subseteq V(G)$, we let

$$\eta[G, W] = \min \{o(F) \mid F \text{ is a routerless connection forest of } G \text{ for } W\}.$$

As a degenerate case, we define $\eta[G, \emptyset] = 0$. We note that $\eta[G, W] = 1$ if and only if G admits a connection tree of G for W such that $\mathsf{R}(T) = \emptyset$. In particular, for $|W| \geq 3$, $\eta[G, W] = 1$ if and only if $G[W]$ is connected.

Lemma 7 ($\star$). *Let G be a cograph and $W \subseteq V(G)$ be a terminal set. If G is the disjoint union of $k \geq 2$ cographs $G_1, \ldots, G_k$, then*

$$\eta[G, W] = \sum_{i \in \{1,\ldots,k\}} \eta[G_i, V(G_i) \cap W].$$

Lemma 8 ($\star$). *Let G be a cograph and $W \subseteq V(G)$ be a terminal set. If G is the join of $k \geq 2$ cographs $G_1, \ldots, G_k$ and there exists a unique $i \in \{1, \ldots, k\}$ such that $V(G_i) \cap W \neq \emptyset$, then*

$$\eta[G, W] = \max \{1, \eta[G_i, W] - n + n_i\},$$

where $n = |V(G)|$ and $n_i = |V(G_i)|$.

Theorem 2. *TCP is linear-time solvable on cographs.*

Proof. Let $I = (G, W, \ell, r)$ be an instance of TCP, where G is a cograph. Assume without loss of generality that $|W| \geq 3$, G is connected but $G[W]$ is not connected. Moreover, based on Lemma 5 and on Corollary 1, assume that $r = 0$ and $\ell \geq o(G[W])$, respectively. Then, let $\mathcal{T}_G$ be the cotree associated with G. Compute $\eta[G, W]$ in a bottom-up manner, according to the post-order traversal of $\mathcal{T}_G$, following the rules described below:

$$\eta[G, W] = \begin{cases} \left[\begin{array}{l} \textbf{case 1.}\ |V(G)| = 1: \\ \ \ 0 \quad \text{if } V(G) \cap W = \emptyset, \\ \ \ 1 \quad \text{otherwise;} \end{array}\right. \\ \\ \left[\begin{array}{l} \textbf{case 2.}\ G = G_1 \cup \cdots \cup G_k,\ \text{for some } k \geq 2: \\ \ \ \sum_{i \in \{1,\ldots,k\}} \eta[G_i, V(G_i) \cap W]; \end{array}\right. \\ \\ \left[\begin{array}{l} \textbf{case 3.}\ G = G_1 \wedge \cdots \wedge G_k,\ \text{for some } k \geq 2: \\ \ \ 0 \quad \text{if } \forall\, i \in \{1, \ldots, k\},\ V(G_i) \cap W = \emptyset, \\ \ \ 1 \quad \text{if } \exists\, i, j \in \{1, \ldots, k\},\ i \neq j,\ V(G_i) \cap W \neq \emptyset \text{ and } V(G_j) \cap W \neq \emptyset, \\ \ \ \max\{1, \eta[G_i, W] - n + n_i\} \quad \text{if } \exists!\, i \in \{1, \ldots, k\},\ V(G_i) \cap W \neq \emptyset, \\ \text{where } n = |V(G)| \text{ and } n_i = |V(G_i)|. \end{array}\right. \end{cases}$$

Return that I is a yes-instance of TCP if and only if $\eta[G, W] = 1$. It is not hard to check that $\eta[G, W]$ can be computed in time linear in the number of vertices and the number of edges of G. The correctness of the rules described above follows from Lemmas 7 and 8. $\square$

4 Bounded Maximum Degree

In this section, we analyse the complexity of TCP when restricted to graphs of bounded maximum degree. More specifically, we prove that TCP remains NP-complete on graphs of maximum degree 3 even if either the parameter $\ell \geq 0$ or the parameter $r \geq 0$ is fixed. In particular, for fixed $r \geq 0$, we show that TCP is NP-complete on graphs of maximum degree 3 that are *planar*.

It is worth mentioning that, if the input graph G is connected and has maximum degree at most 2, then G is either a path or a cycle, and consequently TCP can be trivially solved in polynomial-time, regardless of ℓ or r. Thus, we obtain that our results establish an NP-*complete versus polynomial-time dichotomy* for TCP with respect to the maximum degree of the input graph.

Another interesting fact about our results is that they separate the complexity of TCP from the complexity of its strict variant, so-called S-TCP. Indeed, while we prove that, for each fixed $\ell \geq 0$, TCP is NP-complete on graphs of maximum degree 3, S-TCP was proved to be polynomial-time solvable on graphs of maximum degree 3 if $\ell \geq 0$ is fixed [23].

4.1 Fixed Number of Linkers

First, we consider the case in which the parameter $\ell \geq 0$ is fixed. To prove the NP-completeness of this particular case, we present a polynomial-time reduction from an NP-complete (cf. [24]) variant of 3-SAT called 3-SAT(3). The 3-SAT(3) problem has as input a set X of boolean variables and a set $\mathcal{C}$ of clauses over X that satisfies the following conditions: each clause in $\mathcal{C}$ has two or three distinct literals and each variable in X appears exactly twice positive and once negative in the clauses belonging to $\mathcal{C}$; and it asks whether there exists a truth assignment $\alpha\colon X \to \{\mathit{false}, \mathit{true}\}$ for the variables in X such that every clause in $\mathcal{C}$ has at least one true literal under α.

Construction 3. (Reduction from 3-SAT(3) to TCP on Graphs of Maximum Degree 3). Let $I = (X, \mathcal{C})$ be an instance of 3-SAT(3), with variable set $X = \{x_1, x_2, \ldots, x_p\}$ and clause set $\mathcal{C} = \{C_1, C_2, \ldots, C_q\}$, and let ℓ be a non-negative integer. We let G be the graph obtained from I and ℓ as follows:

- Create the vertices $u_1, u_2, \ldots, u_\ell$ and, for each $i \in \{1, 2, \ldots, \ell - 1\}$, add the edges $u_i u_{i+1}$; moreover, create the vertices w_I and v_I and add the edges $w_I u_1$ and $u_\ell v_I$, originating the path $P_I = \langle w_I, u_1, \ldots, u_\ell, v_I \rangle$;
- For each variable $x_i \in X$, create the gadget G_i such that

$$V(G_i) = \{w_i^1, w_i^2, t_i^1, t_i^2, f_i\} \text{ and } E(G_i) = \{w_i^1 t_i^1, t_i^1 t_i^2, t_i^2 w_i^2, w_i^2 f_i, f_i w_i^1\};$$

- Create a complete binary tree T_I, rooted at v_I, whose leaves are the vertices $w_1^1, \ldots, w_p^1$;
- For each clause $C_j \in \mathcal{C}$, create the vertices v_j^1, v_j^2 and v_j^3, and add the edges $v_j^1 v_j^2$, $v_j^2 v_j^3$ and $v_j^3 v_j^1$;

– For each clause $C_j \in \mathcal{C}$, add the edge $t_i^a v_j^b$ if the b-th literal belonging to C_j corresponds to the a-th occurrence in I of the positive literal x_i, for $x_i \in X$, $a \in \{1, 2\}$ and $b \in \{1, \dots, |C_i|\}$; on the other hand, add the edge $f_i v_j^b$ if the b-th literal belonging to C_j corresponds to the (only) occurrence in I of the negative literal $\overline{x}_i$, for $x_i \in X$ and $b \in \{1, \dots, |C_j|\}$.

We let $g(I, \ell) = (G, W, \ell, r)$ be the instance of TCP such that $W = \{w_I\} \cup V(T_I) \cup \{w_i^1, w_i^2 \mid x_i \in X\} \cup \{v_j^1, v_j^2, v_j^3 \mid C_j \in \mathcal{C}\}$ and $r = 2p$.

Lemma 9. *Let $I = (X, \mathcal{C})$ be an instance of 3-SAT(3). For each $\ell \geq 0$, I is a* yes*-instance of 3-SAT(3) if and only if the instance $g(I, \ell)$ described in Construction 3 is a* yes *instance of TCP.*

Proof. Assume that $X = \{x_1, x_2, \dots, x_p\}$ and $\mathcal{C} = \{C_1, C_2, \dots, C_q\}$. Additionally, assume that $g(I, \ell) = (G, W, \ell, r)$.

First, suppose that there exists a truth assignment $\alpha\colon X \to \{\mathit{false}, \mathit{true}\}$ for the variables in X such that every clause belonging to $\mathcal{C}$ has at least one true literal under α. Then, let S be the vertex set defined as follows

$$\begin{aligned} S = \{t_i^1, t_i^2 \ &\mid x_i \in X, \alpha(x_i) = \mathit{true}\} \cup \{f_i \mid x_i \in X, \alpha(x_i) = \mathit{false}\} \\ &\cup \{w_1^i, w_i^2 \mid x_i \in X\} \cup \{v_j^1, v_j^2, v_j^3 \mid C_j \in \mathcal{C}\} \cup V(P_I) \cup V(T_I), \end{aligned}$$

and let $G[S]$ be the subgraph of G induced by S. We note that $G[S]$ is connected but may contain cycles. Thus, let T be a spanning tree subgraph of $G[S]$ that contains all edges of $G[S]$ except for possibly not containing some edges between the vertices v_j^1, v_j^2 and v_j^3, for $C_j \in \mathcal{C}$. In other words, T is a spanning tree subgraph of $G[S]$ such that $E(T) \supseteq E(G[S]) \setminus \{v_j^a v_j^b \mid a, b \in \{1, 2, 3\}, C_j \in \mathcal{C}\}$. It is not hard to check that T is a connection tree of G for W with linker set $\mathsf{L}(T) = \{u_1, \dots, u_\ell\}$ and router set $\mathsf{R}(T) = \{t_i^1, t_i^2 \mid x_i \in X, \alpha(x_i) = \mathit{true}\} \cup \{f_i \mid x_i \in X, \alpha(x_i) = \mathit{false}\}$. Therefore, $g(I, \ell)$ is a yes-instance of TCP.

Conversely, suppose that $g(I, \ell)$ is a yes-instance of TCP, and let T be a connection tree of G for W such $|\mathsf{L}(T)| \leq \ell$ and $|\mathsf{R}(T)| \leq 2p$. We note that the path P_I must be in T, since every path of G between the terminal vertex w_I and any other terminal vertex $w \in W \setminus \{w_I\}$ contains all the vertices of P_I. Consequently, the graph $T' = T - P_I$ cannot contain any liker, and all non-terminal vertices of T' must be routers. This, along with the fact that $\Delta(G) = 3$, implies that $N_T(v) = N_G(v)$ for each $v \in V(T') \setminus W$. Hence, if $t_i^1 \in V(T)$ or $t_i^2 \in V(T)$, then $w_i^1, t_i^2 \in N_T(t_i^1)$ and $w_i^2, t_i^1 \in N_T(t_i^2)$. Analogously, if $f_i \in V(T)$, then $w_i^1, w_i^2 \in N_T(f_i)$. Thus, since T is acyclic, we have that, for each $x_i \in X$, either $t_i^1, t_i^2 \in V(T)$ and $f_i \notin V(T)$, or $t_i^1, t_i^2 \notin V(T)$ and $f_i \in V(T)$. Then, we define a truth assignment $\alpha\colon X \to \{\mathit{false}, \mathit{true}\}$ for the variables in X as follows: for each $x_i \in X$, $\alpha(x_i) = \mathit{false}$ if and only if $f_i \in V(T)$. We note that, for each $C_j \in \mathcal{C}$, every path of G between the terminal vertices v_j^1, v_j^2, v_j^3 and any other terminal vertex $w \in W \setminus \{v_j^1, v_j^2, v_j^3\}$ must contain one of the vertices t_i^1, t_i^2, f_i for some $x_i \in X$. Moreover, by supposition, $V(T) \supseteq W \supseteq \{v_j^1, v_j^2, v_j^3 \mid C_j \in \mathcal{C}\}$. Consequently, every clause in $\mathcal{C}$ has at least one true literal under α. Therefore, I is a yes-instance of 3-SAT(3). $\square$

Theorem 3 **($\star$).** *For each $\ell \geq 0$, TCP remains* NP*-complete when restricted to graphs of maximum degree* 3.

4.2 Fixed Number of Routers

Now, we consider the case in which the parameter $r \geq 0$ is fixed. To prove the NP-completeness of this particular case, we present a polynomial-time reduction from Hamiltonian cycle on graphs of maximum degree 3, which was shown to be NP-complete by Itai et al. [16]. Hamiltonian cycle has as input a graph G and asks whether G contains a *Hamiltonian cycle*, i.e. a cycle that contains all vertices of G. More precisely, our reduction is actually from the st-Hamiltonian path problem on planar graphs of maximum degree 3, and it is slightly similar to the one described in Construction 2 so as to prove that TCP is NP-complete on strongly chordal graphs. Thus, based on the fact that Hamiltonian cycle is NP-complete on graphs of maximum degree 3, we first prove in the next propositions that st-Hamiltonian path remains NP-complete if the input graph G has maximum degree 3 and s and t have degree 1 in G.

Proposition 2. Hamiltonian cycle *remains* NP*-complete when restricted to planar graphs of maximum degree* 3 *that have at least two adjacent vertices of degree* 2 *each.*

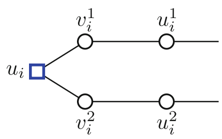
Proof. Itai et al. [16] proved that Hamiltonian cycle is NP-complete on planar graphs of maximum degree 3. Based on their proof (see Lemma 2.1 [16]), we can suppose without loss of generality that the input graph G has at least one vertex of degree 2. Thus, let $u \in V(G)$ be such a vertex, and let $e = uv$ be an edge that has u and v as endpoints, for some $v \in V(G) \setminus \{u\}$. Then, we define H as the graph obtained from G by *subdividing* e, i.e. by removing e, adding a new vertex u_e and adding the edges uu_e and $u_e v$. We note that H is a graph of maximum degree 3 that has at least two adjacent vertices, namely u and u_e, of degree 2 each. Furthermore, it is immediate that G has a Hamiltonian cycle if and only if H has a Hamiltonian cycle.

Proposition 3. st-Hamiltonian path *remains* NP*-complete when restricted to planar graphs of maximum degree* 3 *in which* s *and* t *have degree* 1 *each.*

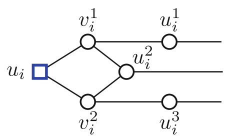
Proof. Let G be a planar graph of maximum degree 3. Based on Proposition 2, assume without loss of generality that G contains two vertices $u, v \in V(G)$ such that $uv \in E(G)$ and $d_G(u) = d_G(v) = 2$. Then, let H be the graph obtained from G by adding two new vertices s and t, and by adding the edges su and vt. We note that H is a graph of maximum degree 3 and that s and t have degree 1 in H each. Furthermore, it is straightforward that G has a Hamiltonian cycle if and only if H has a st-Hamiltonian path. $\square$

Construction 4. (Reduction from st-Hamiltonian path to TCP on Planar Graphs of Maximum Degree 3). Let G be a graph of maximum degree 3 and $s, t \in V(G)$ be distinct vertices of G. Based on Proposition 3, assume without loss of generality that $d_G(s) = d_G(t) = 1$. Moreover, assume that every vertex of G different from s and t has degree at least 2, otherwise G would certainly not admit a st-Hamiltonian path. Also, assume that $V(G) = \{u_1, \ldots, u_n\}$, for some positive integer n, where $s = u_1$ and $t = u_n$. Let r be a non-negative integer. For each $u_i \in V(G) \setminus \{s,t\}$, let $\alpha_i \colon N_G(u_i) \to |N_G(u_i)|$ be the bijection such that, for each two distinct vertices $u_{j_1}, u_{j_2} \in N_G(u_i)$, we have that $\alpha_i(u_{j_1}) < \alpha_i(u_{j_2})$ if and only if $j_1 < j_2$. We let G' be the graph obtained from G, s, t and r as follows:

- Add all vertices of G to G';
- For each vertex $u_i \in V(G)$ of degree 2 in G, add new vertices $v_i^1, v_i^2, u_i^1, u_i^2$ and add the edges $u_iv_i^1$, $u_iv_i^2$, $v_i^1u_i^1$ and $v_i^2u_i^2$ (see Fig. 1b);
- For each vertex $u_i \in V(G)$ of degree 3 in G, add new vertices $v_i^1, v_i^2, u_i^1, u_i^2, u_i^3$ and add the edges $u_iv_i^1$, $u_iv_i^2$, $v_i^1u_i^2$, $v_i^2u_i^2$, $v_i^1u_i^1$ and $v_i^2u_i^3$; (see Fig. 1a)



(a) $d_G(u_i) = 2$



(b) $d_G(u_i) = 3$

Fig. 1. (a) Case in which $d_G(u_i) = 2$: vertices $v_i^1, v_i^2, u_i^1, u_i^2$. (b) Case in which $d_G(u_i) = 3$: vertices $v_i^1, v_i^2, u_i^1, u_i^2, u_i^3$.

- For each vertex $u_i \in V(G)$ and each vertex $u_j \in N_G(u_i)$, add the edges $u_i^a u_j^b$, where $a = \alpha_i(u_j)$ and $b = \alpha_j(u_i)$;
- If $r \geq 1$, create the gadget H_r and the terminal set W_r described in Construction 1, and add the edge $\rho_r s$; if $r = 0$, define $W_r = \emptyset$.

We let $g(G, s, t, r) = (G', W, \ell, r)$ be the instance of TCP such that $W = V(G) \cup W_r$ and $\ell = 4n - 4$.

Lemma 10. *Let G be a graph of maximum degree 3 and $s, t \in V(G)$ be two distinct vertices of G. Assume that s and t have degree 1 in G each. For each $r \geq 0$, G admits a st-Hamiltonian path if and only if the instance $g(G, s, t, r)$ described in Construction 4 is a* yes*-instance of TCP.*

Proof. Assume that $V(G) = \{u_1, \ldots, u_n\}$, for some positive integer n, where $s = u_1$ and $t = u_n$. Additionally, assume that $g(G, s, t, r) = (G', W, \ell, r)$ and, for simplicity, consider $W_r = V(H_r) = E(H_r) = \emptyset$ if $r = 0$.

Suppose that there is in G a Hamiltonian path $P = \langle u_{j_1}, u_{j_2}, \ldots, u_{j_{n-1}}, u_{j_n} \rangle$ such that $s = u_{j_1}$ and $t = u_{j_n}$. Then, let S be the vertex set defined as follows:

$$\begin{aligned} S = V(H_r) \cup V(G) \cup \{v_i^1, v_i^2 \mid i \in \{2, \ldots, n-1\}\} \cup \{u_{j_2}^{\alpha_{j_2}(s)}, u_{j_{n-1}}^{\alpha_{j_{n-1}}(t)}\} \\ \cup \{u_{j_i}^a, u_{j_{i+1}}^b \mid a = \alpha_{j_i}(u_{j_{i+1}}), b = \alpha_{j_{i+1}}(u_{j_i}), i \in \{2, \ldots, n-2\}\}, \end{aligned}$$

and let $G'[S]$ be the subgraph of G' induced by S. We note that $G'[S]$ is connected but may contain cycles. More precisely, every cycle of $G'[S]$ is of the form $\langle u_i, v_i^1, u_i^2, v_i^2, u_i \rangle$, and it exists if and only if $d_G(u_i) = 3$ and either $S \supseteq \{u_i^1, u_i^2\}$ or $S \supseteq \{u_i^2, u_i^3\}$, for $u_i \in V(G) \setminus \{s,t\}$. Thus, we let T be the graph obtained from $G'[S]$ by removing, for each vertex $u_i \in V(G) \setminus \{s,t\}$ with $d_G(u_i) = 3$, the edge $v_i^1 u_i^2$ if $S \supseteq \{u_i^1, u_i^2\}$, or the edge $v_i^2 u_i^2$ if $S \supseteq \{u_i^2, u_i^3\}$. One can verify that T is a connection tree of G' for W such that $\mathsf{L}(T) = S \setminus (V(H_r) \cup V(G))$ and $\mathsf{R}(T) = \{\rho_1, \ldots, \rho_r\}$. Therefore, $g(G,s,t,r)$ is a yes-instance of TCP.

Conversely, suppose that $g(G,s,t,r)$ is a yes-instance of TCP, and let T be a connection tree of G for W such that $|\mathsf{L}(T)| \leq 4n-4$ and $|\mathsf{R}(T)| \leq r$. We note that $\mathsf{R}(T) = \{\rho_1, \ldots, \rho_r\}$. Consequently, $T' = T - H_r$ cannot contain any router, and all non-terminal vertices of T' must be linkers. Moreover, by construction, s and t have degree 1 in T'. This implies that the vertices $u_2, \ldots, u_{n-1}$ have degree exactly 2 in T' each, otherwise T would not be connected or $W \not\subseteq V(T)$. Hence, T' consists in a path P' between s and t of the form

$$P' = \langle s, u_{j_2}^{a_2}, v_{j_2}^{c_2}, u_{j_2}, v_{j_2}^{c'_2}, u_{j_2}^{b_2}, \ldots, u_{j_{n-1}}^{a_{n-1}}, v_{j_{n-1}}^{c_{n-1}}, u_{j_{n-1}}, v_{j_{n-1}}^{c'_{n-1}}, u_{j_{n-1}}^{b_{n-1}}, t \rangle,$$

where $a_i = \alpha_{j_i}(u_{j_{i+1}})$, $b_i = \alpha_{j_{i+1}}(u_{j_i})$, and $c_i, c'_i \in \{1,2\}$, with $c_i \neq c'_i$ for each $i \in \{2, \ldots, n-2\}$. Therefore, G admits a st-Hamiltonian path. Indeed, one can verify that $\langle s, u_{j_2}, \ldots, u_{j_{n-1}}, t \rangle$ is a st-Hamiltonian path of G. $\square$

Theorem 4 ($\star$). *For each $r \geq 0$, TCP remains NP-complete when restricted to planar graphs of maximum degree 3.*

5 Concluding Remarks

We conclude this work by posing some open questions. As mentioned in the introduction, if Steiner tree is NP-complete on a graph class $\mathcal{G}$, then, unless $\mathsf{P} = \mathsf{NP}$, TCP cannot be solved in polynomial-time on $\mathcal{G}$. Nevertheless, possibly TCP is polynomial-time solvable on a graph class $\mathcal{G}$ if either $\ell \geq 0$ or $r \geq 0$ is fixed, while Steiner tree remains NP-complete on $\mathcal{G}$. Motivated by this, we ask for the existence of such graph classes. Another interesting question concerns the complexity of TCP when the number of terminal vertices is fixed. Even though it is well-known that Steiner tree can be solved in polynomial-time if the number of terminal vertices is fixed [11], the complexity of TCP in this particular case has not been settled yet. Finally, we remark that, beyond cographs, Steiner tree is also polynomial-time solvable on the superclass of permutation graphs [4] and on the superclass of graphs of constant cliquewidth [1]. However, it is unknown whether TCP admits polynomial-time algorithms when restricted to such graph classes.

Acknowledgments. The authors would like to thank anonymous reviewers for their suggestions and comments. This work was supported by the Brazilian agencies CAPES (Finance Code: 001), CNPq (Grant Numbers: CNPq/GD 140399/2017-8, 407635/2018-1, 303726/2017-2) and FAPERJ (Grant Numbers: E-26/202.793/2017 and E-26/203.272/2017).

References

1. Bergougnoux, B., Kanté, M.: Fast exact algorithms for some connectivity problems parameterized by clique-width. Theoret. Comput. Sci. **782**, 30–53 (2019)
2. Björklund, A., Husfeldt, T., Kaski, P., Koivisto, M.: Fourier meets möbius: fast subset convolution. In: Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing. p. 67–74. STOC 2007, Association for Computing Machinery, New York, USA (2007)
3. Bondy, A., Murty, U.: Graph Theory. Graduate Texts in Mathematics. Springer, London (2008)
4. Colbourn, C.J., Stewart, L.K.: Permutation graphs: connected domination and Steiner trees. Discrete Math. **86**(1–3), 179–189 (1990)
5. Corneil, D.G., Lerchs, H., Burlingham, S.L.: Complement reducible graphs. Discrete Appl. Math. **3**(3), 163–174 (1981)
6. Corneil, D.G., Perl, Y., Stewart, L.K.: A linear recognition algorithm for cographs. SIAM J. Comput. **14**(4), 926–934 (1985)
7. Cygan, M., Pilipczuk, M., Pilipczuk, M., Wojtaszczyk, J.O.: Kernelization hardness of connectivity problems in d-degenerate graphs. Discrete Appl. Math. **160**(15), 2131–2141 (2012)
8. D'Atri, A., Moscarini, M.: Distance-hereditary graphs, Steiner trees, and connected domination. SIAM J. Comput. **17**(3), 521–538 (1988)
9. Dourado, M.C., Oliveira, R.A., Protti, F., Souza, U.S.: Design of connection networks with bounded number of non-terminal vertices. In: Proceedings of V Latin-American Workshop on Cliques in Graphs. Matemática Contemporânea, vol. 42, pp. 39–47. SBM, Buenos Aires (2014)
10. Dourado, M.C., Oliveira, R.A., Protti, F., Souza, U.S.: Conexão de terminais com número restrito de roteadores e elos. In: proccedings of XLVI Simpósio Brasileiro de Pesquisa Operacional. pp. 2965–2976 (2014)
11. Dreyfus, S.E., Wagner, R.A.: The Steiner problem in graphs. Networks **1**(3), 195–207 (1971)
12. Farber, M.: Characterizations of strongly chordal graphs. Discrete Math. **43**(2), 173–189 (1983)
13. Garey, M.R., Johnson, D.S.: The rectilinear Steiner tree problem is NP-complete. SIAM J. Appl. Math. **32**(4), 826–834 (1977)
14. Gargano, L., Hammar, M., Hell, P., Stacho, L., Vaccaro, U.: Spanning spiders and light-splitting switches. Discrete Math. **285**(1), 83–95 (2004)
15. Hwang, F.K., Richards, D.S., Winter, P.: The Steiner tree problem, Annals of Discrete Mathematics, vol. 53. Elsevier (1992)
16. Itai, A., Papadimitriou, C.H., Szwarcfiter, J.L.: Hamilton paths in grid graphs. SIAM J. Comput. **11**(4), 676–686 (1982)
17. Karp, R.M.: Reducibility Among Combinatorial Problems, pp. 85–103. Springer, Boston (1972). https://doi.org/10.1007/978-1-4684-2001-2_9
18. Lin, G., Xue, G.: On the terminal Steiner tree problem. Inf. Proces. Lett. **84**(2), 103–107 (2002)
19. Lozzo, G.D., Rutter, I.: Strengthening hardness results to 3-connected planar graphs (2016). https://arxiv.org/abs/1607.02346
20. Lu, C.L., Tang, C.Y., Lee, R.C.T.: The full Steiner tree problem. Theoret. Comput. Sci. **306**(1–3), 55–67 (2003)
21. Melo, A.A., Figueiredo, C.M.H., Souza, U.S.: On undirected two-commodity integral flow, disjoint paths and strict terminal connection problems. Networks (accepted for publication)

22. Melo, A.A., Figueiredo, C.M.H., Souza, U.S.: Connecting terminals using at most one router. In: proceedings of VII Latin-American Workshop on Cliques in Graphs. Matemática Contemporânea, vol. 45, pp. 49–57. SBM (2017)
23. Melo, A.A., Figueiredo, C.M.H., Souza, U.S.: A multivariate analysis of the strict terminal connection problem. J. Comput. Syst. Sci. **111**, 22–41 (2020)
24. Müller, H.: Hamiltonian circuits in chordal bipartite graphs. Discr. Math. **156**(1–3), 291–298 (1996)
25. Müller, H., Brandstädt, A.: The NP-completeness of Steiner tree and dominating set for chordal bipartite graphs. Theoret. Comput. Sci. **53**(2–3), 257–265 (1987)
26. Nederlof, J.: Fast polynomial-space algorithms using inclusion-exclusion. Algorithmica **65**(4), 868–884 (2013)
27. Watel, D., Weisser, M.-A., Bentz, C., Barth, D.: Steiner problems with limited number of branching nodes. In: Moscibroda, T., Rescigno, A.A. (eds.) SIROCCO 2013. LNCS, vol. 8179, pp. 310–321. Springer, Cham (2013). https://doi.org/10.1007/978-3-319-03578-9_26
28. Watel, D., Weisser, M.-A., Bentz, C., Barth, D.: Directed Steiner trees with diffusion costs. J. Comb. Optim. **32**(4), 1089–1106 (2015). https://doi.org/10.1007/s10878-015-9925-3
29. White, K., Farber, M., Pulleyblank, W.: Steiner trees, connected domination and strongly chordal graphs. Networks **15**(1), 109–124 (1985)