Theoretical Computer Science 867 (2021) 30–39

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Steiner trees for hereditary graph classes: A treewidth perspective ☆

Hans L. Bodlaender [a], Nick Brettell [c,∗], Matthew Johnson [b], Giacomo Paesani [b], Daniël Paulusma [b], Erik Jan van Leeuwen [a]

[a] *Department of Information and Computing Sciences, Utrecht University, the Netherlands*
[b] *Department of Computer Science, Durham University, UK*
[c] *School of Mathematics and Statistics, Victoria University of Wellington, New Zealand*

A R T I C L E I N F O

Article history:
Received 15 April 2020
Received in revised form 15 December 2020
Accepted 4 March 2021
Available online 10 March 2021
Communicated by F.V. Fomin

Keywords:
Steiner tree
Hereditary graph class
Treewidth

A B S T R A C T

We consider the classical problems (Edge) Steiner Tree and Vertex Steiner Tree after restricting the input to some class of graphs characterized by a small set of forbidden induced subgraphs. We show a dichotomy for the former problem restricted to (H_1, H_2)-free graphs and a dichotomy for the latter problem restricted to H-free graphs. We find that there exists an infinite family of graphs H such that Vertex Steiner Tree is polynomial-time solvable for H-free graphs, whereas there exist only two graphs H for which this holds for Edge Steiner Tree (assuming $\mathsf{P} \neq \mathsf{NP}$). We also find that Edge Steiner Tree is polynomial-time solvable for (H_1, H_2)-free graphs if and only if the treewidth of the class of (H_1, H_2)-free graphs is bounded (subject to $\mathsf{P} \neq \mathsf{NP}$). To obtain the latter result, we determine all pairs (H_1, H_2) for which the class of (H_1, H_2)-free graphs has bounded treewidth.

1. Introduction

Let $G = (V, E)$ be a connected graph and $U \subseteq V$ be a set of *terminal* vertices. A *Steiner tree* for U (of G) is a tree in G that contains all vertices of U. An *edge weighting* of G is a function $w_E : E \to \mathbb{Q}^+$ (where $\mathbb{Q}^+$ denotes the set of strictly positive rational numbers). For a tree T in G, the *edge weight* $w_E(T)$ of T is the sum $\sum_{e \in E(T)} w_E(e)$. We consider the classical problem:

Edge Steiner Tree

Instance: a connected graph $G = (V, E)$ with an weighting w_E, a subset $U \subseteq V$ of terminals and a positive integer k.
Question: does G have a Steiner tree T_U for U with $w_E(T_U) \leq k$?

☆ An extended abstract of this paper appeared in the proceedings of LATIN 2020 [3]. The paper received support from the Leverhulme Trust (RPG-2016-258) and the Royal Society (IES\R1\191223).

∗ Corresponding author.
E-mail addresses: h.l.bodlaender@uu.nl (H.L. Bodlaender), nick.brettell@vuw.ac.nz (N. Brettell), matthew.johnson2@durham.ac.uk (M. Johnson), giacomo.paesani@durham.ac.uk (G. Paesani), daniel.paulusma@durham.ac.uk (D. Paulusma), e.j.vanleeuwen@uu.nl (E.J. van Leeuwen).

https://doi.org/10.1016/j.tcs.2021.03.012
0304-3975/

This is often known simply as Steiner Tree, but we wish to distinguish it from a closely related problem. A *vertex weighting* of G is a function $w_V : V \to \mathbb{Q}^+$. For a tree T in G, the *vertex weight* $w_V(T)$ of T is the sum $\sum_{v \in V(T)} w(v)$. The following problem is sometimes known as Node-Weighted Steiner Tree.

Vertex Steiner Tree

Instance: a connected graph $G = (V, E)$ with a vertex weighting w_V, a subset $U \subseteq V$ and a positive integer k.
Question: does G have a Steiner tree T_U for U with $w_V(T_U) \le k$?

Note that we defined the problem for connected inputs, as for general inputs a Steiner tree exists only if the vertices of U all belong to the same connected component. Note also that Edge Steiner Tree is a generalization of the Spanning Tree problem (set $U = V(G)$). We refer to the textbooks of Du and Hu [13] and Prömel and Steger [23] for further background information on Steiner trees.

We consider the problems Edge Steiner Tree and Vertex Steiner Tree separately so that, for any graph under consideration, we have either an edge or vertex weighting but not both, so we will generally denote weightings by w without any subscript. Moreover, when we use the following terminology there is no ambiguity. We say that a Steiner tree of least possible weight is *minimum*, and that an instance of one of the two problems is *unweighted* if the weighting is constant. We denote instances of the weighted problems by (G, w, U, k) and of the unweighted problems by (G, U, k). It is well known that the unweighted versions of Edge Steiner Tree and Vertex Steiner Tree are NP-complete [19,14]. Moreover, as an n-vertex tree has exactly $n - 1$ edges, one can make the following observation.

Observation 1. *The unweighted versions of* Edge Steiner Tree *and* Vertex Steiner Tree *are equivalent.*

Our Focus We focus on the computational complexity of Edge Steiner Tree and Vertex Steiner Tree for *hereditary* graph classes, which are those graph classes that are closed under vertex deletion. We do this from a *systematic* point of view. It is well known, and readily seen, that a graph class $\mathcal{G}$ is hereditary if and only if it can be characterized by a set $\mathcal{H}$ of forbidden induced subgraphs. That is, a graph G belongs to $\mathcal{G}$ if and only if G has no induced subgraph isomorphic to some graph in $\mathcal{H}$. We normally require $\mathcal{H}$ to be minimal, in which case it is unique and we denote it by $\mathcal{H}_{\mathcal{G}}$. We note that $\mathcal{H}_{\mathcal{G}}$ may have infinite size; for example, if $\mathcal{G}$ is the class of bipartite graphs, then $\mathcal{H}_{\mathcal{G}} = \{C_3, C_5, \dots, \}$, where C_r denotes the cycle on r vertices. For a systematic complexity study of a graph problem, we may first consider *monogenic graph classes* or *bigenic* graph classes, which are classes $\mathcal{G}$ with $|\mathcal{H}_{\mathcal{G}}| = 1$ or $|\mathcal{H}_{\mathcal{G}}| = 2$, respectively. This is the approach we follow here.

Our Results We prove a dichotomy for Edge Steiner Tree for bigenic graph classes in Section 2 and a dichotomy for Vertex Steiner Tree for monogenic graph classes in Section 3. We relate our first dichotomy to a dichotomy for boundedness of *treewidth* of bigenic graph classes. The parameter treewidth, which we formally define in Section 2.2, measures how close a graph is to being a tree. If we wish to solve a problem on some graph class $\mathcal{G}$ and we know that $\mathcal{G}$ has small treewidth, then we can try to mimic efficient algorithms for trees to obtain efficient algorithms for $\mathcal{G}$. Many discrete optimization problems can be solved in polynomial time on every graph class of bounded treewidth. The Edge Steiner Tree problem is an example of such a problem (see, for instance, [9] or, for a faster algorithm [4]).

In order to describe our dichotomies we need to introduce some extra terminology. We denote the *disjoint union* of two vertex-disjoint graphs G and H by $G + H = (V(G) \cup V(H), E(G) \cup E(H))$, and the disjoint union of s copies of G by sG. A *linear forest* is a disjoint union of paths. For a graph H, a graph is H-*free* if it has no induced subgraph isomorphic to H. For a set of graphs $\{H_1, \dots, H_p\}$, a graph is $(H_1, \dots, H_p)$-*free* if it is H_i-free for every $i \in \{1, \dots, p\}$. We let K_r and P_r denote the complete graph and path on r vertices, respectively. The *complete bipartite* graph $K_{s,t}$ is the graph whose vertex set can be partitioned into two sets S and T of size s and t, respectively, such that for any two distinct vertices u, v, we have $uv \in E$ if and only if $u \in S$ and $v \in T$ or vice versa. We call $K_{1,3}$ the *claw*.

Theorem 1. *Let H_1 and H_2 be two graphs. If one of the following cases holds:*

1. *$H_1 = K_r$ or $H_2 = K_r$ for some $r \in \{1, 2\}$;*
2. *$H_1 = K_3$ and $H_2 = K_{1,3}$, or vice versa;*
3. *$H_1 = K_r$ for some $r \ge 3$ and $H_2 = P_3$, or vice versa; or*
4. *$H_1 = K_r$ for some $r \ge 3$ and $H_2 = sP_1$ for some $s \ge 1$, or vice versa,*

then the treewidth of the class of (H_1, H_2)-free graphs is bounded and Edge Steiner Tree *is polynomial-time solvable for (H_1, H_2)-free graphs. In all other cases, the treewidth of (H_1, H_2)-free graphs is unbounded and* Edge Steiner Tree *is* NP-*complete for (H_1, H_2)-free graphs.*

Theorem 2. *Let H be a graph. If H is an induced subgraph of $sP_1 + P_4$ for some $s \ge 0$, then* Vertex Steiner Tree *is polynomial-time solvable for H-free graphs, otherwise even unweighted* Vertex Steiner Tree *is* NP-*complete.*

We make the following observations about these two results:

1. In Theorem 1 we show that Edge Steiner Tree can be solved in polynomial time for (H_1, H_2)-free graphs if and only if the treewidth of the class of (H_1, H_2)-free graphs is bounded (assuming $\mathsf{P} \neq \mathsf{NP}$). However, such a 1-to-1 correspondence holds neither between Vertex Steiner Tree and treewidth, nor between Vertex Steiner Tree and the less restrictive width parameter mim-width. This can be seen as follows. It is known that Vertex Steiner Tree is polynomial-time solvable for a graph class of bounded mim-width provided that a branch decomposition of constant mim-width can be found in polynomial time for the class [1]. In particular, every graph class of bounded treewidth has this property. However, complete graphs, and hence P_4-free graphs, have unbounded treewidth, whereas co-bipartite graphs, and hence $3P_1$-free graphs, have unbounded mim-width [6]. In Section 4 we discuss this connection between Edge Steiner Tree and treewidth further. In the same section we also pose a number of open problems.

2. Theorem 1 also provides a dichotomy for boundedness of treewidth of (H_1, H_2)-free graphs. For the less restrictive width parameter clique-width (or equivalently boolean-width, rank-width, module-width, or NLC-width [8,18,22,24]) or the even less restrictive width parameter mim-width, such dichotomies have not yet been established for (H_1, H_2)-free graphs. We refer to [12] and [7] for state-of-the-art summaries for clique-width and mim-width, respectively.

3. The restriction of Theorem 1 to monogenic graph classes, that is, taking $H = H_1 = H_2$, yields only two (trivial) graphs H, namely $H = K_1$ or $H = K_2$, for which the restriction of Edge Steiner Tree to H-free graphs can be solved in polynomial time. In contrast, by Theorem 2, Vertex Steiner Tree can, when restricted to H-free graphs, be solved in polynomial time for an infinite family of linear forests H, namely $H = sP_1 + P_4$ $(s \geq 0)$.

4. Theorem 2 is also a dichotomy for the unweighted Vertex Steiner Tree problem. Moreover, as the unweighted versions of Edge Steiner Tree and Vertex Steiner Tree are equivalent by Observation 1, Theorem 2 is also a classification of the unweighted version of Edge Steiner Tree.

2. Proof of Theorem 1

In this section we give a proof for our first dichotomy, which is for Edge Steiner Tree for (H_1, H_2)-free graphs. We note that this is not the first systematic study of Edge Steiner Tree. For example, Renjitha and Sadagopan [25] proved that unweighted Edge Steiner Tree is NP-complete for $K_{1,5}$-free split graphs, but can be solved in polynomial time for $K_{1,4}$-free split graphs. We present a number of other results from the literature, which we collect in Section 2.1, together with some lemmas that follow from these results. Then in Section 2.2 we discuss the notion of treewidth; as mentioned, this notion will play an important role. We then use these results to prove Theorem 1 in Section 2.3.

2.1. Preliminaries

The NP-completeness of Edge Steiner Tree on complete graphs follows from the result [19] that the general problem is NP-complete: to obtain a reduction add any missing edges and give them sufficiently large weight such that they will never be used in any solution. Bern and Plasman proved the following stronger result.

Lemma 1 *([2]).* Edge Steiner Tree *is* NP-*complete for complete graphs where every edge has weight* 1 *or* 2.

To subdivide an edge $e = uv$ means to delete e and add a vertex w and edges uw and vw. Let r be a positive integer. To say that e is subdivided r times means that e is replaced by a path $P_e = uw_1 \cdots w_r v$ of $r + 1$ edges. The r-*subdivision* of a graph H is the graph obtained from H after subdividing each edge exactly r times. If we say that a graph is a *subdivision* of H, then we mean it can be obtained from H using subdivisions (the number of subdivisions can be different for each edge and some edges might not be subdivided at all). A graph G contains a graph H as a *subdivision* if G contains a subdivision of H as a subgraph.

Proposition 1. *If* Edge Steiner Tree *is* NP-*complete for a class* $\mathcal{C}$ *of graphs, then for every* $r \geq 0$*, it is also* NP-*complete for the class of* r-*subdivisions of graphs in* $\mathcal{C}$.

Proof. Let (G, w, U, k) be an instance of Edge Steiner Tree where $G \in \mathcal{C}$. Let G' be the r-subdivision of G and for each edge e in G, let P_e be the corresponding path on $r + 1$ edges in G'. We define an edge weighting w' for G' by letting $w'(e') = w(e)/(r + 1)$ for each $e \in E(G)$ and for each $e' \in E(P_e)$. In any minimum Steiner tree T' for U of (G', w'), for each $e \in E$, either all or no edges of P_e are in T'. Then it is easy to see that there is a bijection that preserves weight between minimum Steiner trees of (G, w) for U and minimum Steiner trees of (G', w') for U. □

We make the following observation.

Lemma 2. Edge Steiner Tree *is* NP-*complete for complete bipartite graphs.*

Fig. 1. A wall of height 2 and the net-wall obtained by applying a wye-net transformation.

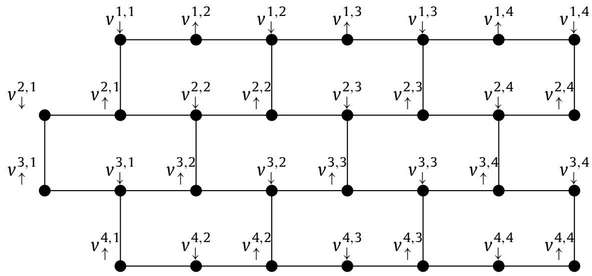

Fig. 2. A wall of height 3 (obtained from a grid with $n = 4$ and $m = 4$) and the labelling of its vertices per Lemma 3. Note that $v^{n,1}$ and $v^{1,1}$ are exceptional.

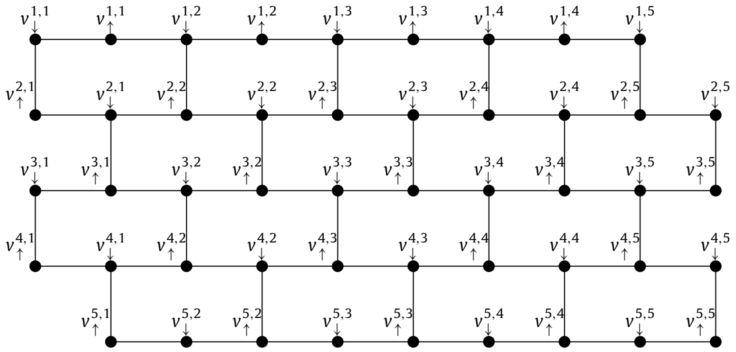

Fig. 3. A wall of height 4 (obtained from a grid with $n = 5$ and $m = 5$) and the labelling of its vertices per Lemma 3. Note that $v^{n,1}$ and $v^{1,m}$ are exceptional.

Proof. As Edge Steiner Tree is NP-complete in general, and the 1-subdivision of any graph is bipartite, the problem remains NP-complete for bipartite graphs by Proposition 1. We describe a reduction from the problem on bipartite graphs to the problem on complete bipartite graphs. Let (G, w, U, k) be an instance of Edge Steiner Tree where G is bipartite. Let M be the sum of the weights of all the edges of G and note that if $k > M$ we can redefine it as $k = M$ without changing the problem. Let G' be the complete bipartite graph obtained from G by adding all missing edges that preserve the bipartition. Let w' be an edge weighting for G' where $w'(e) = w(e)$ if e belongs to G and $w'(e) = M + 1$ otherwise. Thus (G', w', U, k) is an instance of Edge Steiner Tree for complete bipartite graphs and clearly G has a Steiner tree T_U of weight at most k if and only if G' has a Steiner tree T'_U of weight at most k. $\square$

The next theorem follows by inspection of the reduction of Garey and Johnson for Rectilinear Steiner Tree [16]. Let n and m be positive integers. An $n \times m$ grid graph has vertex set $\{v^{i,j} \mid 1 \le i \le n, 1 \le j \le m\}$ and $v^{i,j}$ has neighbours $v^{i-1,j}$ (if $i > 1$), $v^{i+1,j}$ (if $i < n$), $v^{i,j-1}$ (if $j > 1$), and $v^{i,j+1}$ (if $j < m$). Think of $v^{1,1}$ as the top-left corner of the grid, and in $v^{i,j}$, i indicates the row of the grid containing the vertex, while j indicates the column.

Theorem 3 *([16]). Unweighted* Edge Steiner Tree *is* NP-*complete for grid graphs.*

A *wall* is a graph which can be thought of as a hexagonal grid. See Fig. 1, 2, and 3 for three examples of walls of different *heights* and *widths*. We refer to [11] for a formal definition.

From a wall of height h we obtain a *net-wall* by doing the following: for each wall vertex u with three neighbours v_1, v_2, v_3, replace u and its incident edges with three new vertices u_1, u_2, u_3 and edges u_1v_1, u_2v_2, u_3v_3, u_1u_2, u_1u_3, u_2u_3. We call this a *wye-net transformation*, reminiscent of the well-known wye-delta transformation (see [27]). Note that a net-wall is $K_{1,3}$-free but contains an induced *net*, which is the graph obtained from a triangle on vertices a_1, a_2, a_3 and three new vertices b_1, b_2, b_3 by adding the edge a_ib_i for $i = 1, 2, 3$. We have two results related to these classes.

Lemma 3. *For every $r \ge 0$,* Edge Steiner Tree *is* NP-*complete for r-subdivisions of walls.*

Proof. We reduce from unweighted Edge Steiner Tree on grid graphs, which is NP-complete by Theorem 3. Let (G, U, k) be an instance of unweighted Edge Steiner Tree where G is an $n \times m$ grid graph. By adding a few rows and columns to the side of the grid, we may assume that the two rows and columns forming the boundary grid are free of terminals and that the outer row and column will not be used by any optimal solution to (G, U, k). We call such an instance *clean*. We call a Steiner tree *neat* if it avoids the outer row and column of the grid or the outer row and column of the wall we construct in the reduction. (The outer row and column of a wall means those vertices corresponding to the outer row and column of the grid from which it is obtained by the splitting of the vertices as explained below.) Note that, because the instance is clean, asking for a neat Steiner tree is not a restriction.

From G, we obtain a graph W as follows. See Fig. 2 and 3 for examples. Two vertices of G are exceptional: $v^{n,1}$ is always exceptional, $v^{1,m}$ is exceptional if n is odd, and $v^{1,1}$ is exceptional if n is even. For every vertex $v^{i,j}$ of G that is not exceptional, W contains vertices $v^{i,j}_{\uparrow}$ and $v^{i,j}_{\downarrow}$ that are joined by an edge. We call these edges *new*. We also add to W vertices $v^{n,1}_{\uparrow}$, and $v^{1,m}_{\downarrow}$ (if $v^{1,m}$ is exceptional) or $v^{1,1}_{\downarrow}$ (if $v^{1,1}$ is exceptional). We add an edge from $v^{i,j}_{\downarrow}$ to $v^{i+1,j}_{\uparrow}$, for $1 \le i \le n-1$, $1 \le j \le m$. For $1 \le i \le n$, $1 \le j \le m-1$, if i is odd and n is even or if i is even and n is odd, we add an edge from $v^{i,j}_{\downarrow}$ to $v^{i,j+1}_{\uparrow}$, and, otherwise, we add an edge from $v^{i,j}_{\uparrow}$ to $v^{i,j+1}_{\downarrow}$. The edges that are not new are *original*.

We note that W is a wall obtained from G by splitting each vertex in two (except the exceptional vertices that lie in a corner of the grid), and that there is a bijection between the original edges of W and the edges of G. We define an edge weighting w' for W by letting the weight of each original edge be 1 and the weight of each new edge be a rational number ε, where $\varepsilon > 0$ is chosen so that the sum of the weights of all new edges is less than 1. We define a set of terminals U' for W: if $v^{i,j}$ is in U, then U' contains each of $v^{i,j}_{\downarrow}$ and $v^{i,j}_{\uparrow}$ (note that both vertices exist because the instance is clean). Observe that W is also clean.

We claim that there is a neat Steiner tree of k edges in G for terminal set U if and only if there is a neat Steiner tree of weight $k + (k+1)\varepsilon$ in (W, w') for terminal set U'. Indeed, any neat Steiner tree T in G for terminal set U of k edges corresponds naturally to a neat Steiner tree T' for U' in (W, w') of weight $k + (k+1)\varepsilon$ by adding for each vertex of T the corresponding new edge to T' (note that the neatness of T implies that the corresponding new edges exist). Conversely, any neat Steiner tree T' for U' in (W, w') of weight $k + (k+1)\varepsilon$ corresponds naturally to a neat Steiner tree T for U in G of k edges by removing all new edges from T'. Effectively, this mimics the splitting and contraction operations which can be seen as the way in which we obtain W from G and vice versa.

The lemma now follows immediately from Proposition 1. □

The next lemma has a similar proof.

Lemma 4. *For every $r \ge 0$,* Edge Steiner Tree *is* NP*-complete for r-subdivisions of net-walls.*

Proof. We reduce from Edge Steiner Tree on walls, which is NP-complete by Lemma 3. Consider an instance (W, w, U, k) of Edge Steiner Tree where W is a wall. By the construction of Lemma 3, we may assume that the two outer rows and columns of the wall are free of terminals and the outer row and column will not be used by any optimal solution to (W, w, U, k). We call such an instance *clean*. We call a Steiner tree *neat* if it avoids the outer row and column. Note that, because the instance is clean, asking for a neat Steiner tree is not a restriction. We now apply a wye-net transformation to W to obtain a *net-wall* N.

The edges of the triangles added through the wye-net transformations are called *new* and all other edges of N are *original* (since they admit a bijection to the edges of W). We create an edge weighting w' on N. For each original edge e, $w'(e) = w(e)$. Let s be the smallest weight of an edge of (W, w). Let the weight of each new edge be a rational number ε, where $\varepsilon > 0$ is chosen so that the sum of the weights of all new edges is less than s. We define a set of terminals U' for N. We note that each vertex v of W that is in U corresponds to a set of three vertices of N that form a triangle, by the fact that the instance is clean. If v is in U, then we add to U' all three vertices of the triangle. Note that N is also clean.

We claim that there is a neat Steiner tree of weight k in (W, w) for terminal set U if and only if there is a neat Steiner tree of weight $k + 2(k+1)\varepsilon$ in (N, w') for terminal set U'. Indeed, any neat Steiner tree T in W for terminal set U of weight at most k corresponds naturally to a neat Steiner tree T' for U' in (N, w') of weight $k + 2(k+1)\varepsilon$ adding for each vertex of the $k+1$ vertices of T any two edges of the corresponding triangle to T' (note that the neatness of T implies that the corresponding triangles exist). Conversely, any neat Steiner tree T' for U' in (N, w') of weight $k+2(k+1)\varepsilon$ corresponds naturally to a neat Steiner tree T for U in (W, w) of weight at most k by removing all new edges from T'.

The lemma now follows immediately from Proposition 1. □

2.2. Treewidth and implications

A *tree decomposition* of a graph $G = (V, E)$ is a tree T whose vertices, which are called *nodes*, are subsets of V and has the following properties: for each $v \in V$, the nodes of T that contain v induce a connected subgraph with at least one node, and, for each edge $vw \in E$, there is at least one node of T that contains v and w. We refer to Fig. 4 for an example.

The sets of vertices of G that form the nodes of T are called *bags*. The *width* of T is one less than the size of its largest bag. The *treewidth* of G is the minimum width of its tree decompositions. A graph class $\mathcal{G}$ has *bounded treewidth* if there

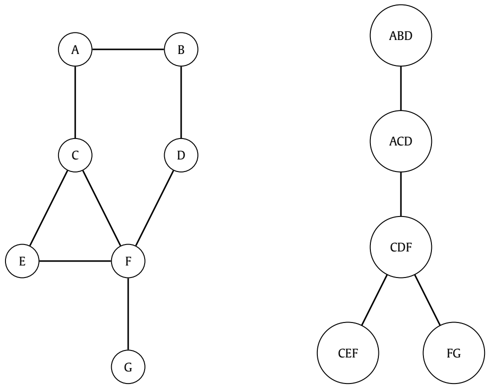

Fig. 4. A graph, and a tree decomposition of the graph with width 2.

exists a constant c such that each graph in $\mathcal{G}$ has treewidth at most c; otherwise $\mathcal{G}$ has *unbounded treewidth*. As trees with at least one edge form exactly the class of connected graphs with treewidth 1, the treewidth of a graph can be seen as a measure that indicates how close a graph is to being a tree.

We recall the following well-known result.

Lemma 5 *([4,9]).* Edge Steiner Tree *can be solved in polynomial time on every graph class of bounded treewidth.*

For some of our poofs we need the well-known Robertson-Seymour Grid-Minor Theorem (also called the Excluded Grid Theorem), which can be formulated for walls.

Theorem 4 *([26]). For every integer h, there exists a constant c_h such that a graph has treewidth at least c_h if and only if it contains a wall of height h as a subdivision.*

We will use two lemmas, both of which follow immediately from Theorem 4.

Lemma 6. *For every $r \geq 0$, the class of r-subdivisions of walls has unbounded treewidth.*

Lemma 7. *For every $r \geq 0$, the class of r-subdivisions of net-walls has unbounded treewidth.*

2.3. The proof

We are now ready to prove Theorem 1.

Theorem 1 *(restated). Let H_1 and H_2 be two graphs. If one of the following cases holds:*

1. *$H_1 = K_r$ or $H_2 = K_r$ for some $r \in \{1, 2\}$;*
2. *$H_1 = K_3$ and $H_2 = K_{1,3}$, or vice versa;*
3. *$H_1 = K_r$ for some $r \geq 3$ and $H_2 = P_3$, or vice versa; or*
4. *$H_1 = K_r$ for some $r \geq 3$ and $H_2 = sP_1$ for some $s \geq 1$, or vice versa,*

then the treewidth of the class of (H_1, H_2)-free graphs is bounded and Edge Steiner Tree *is polynomial-time solvable for (H_1, H_2)-free graphs. In all other cases, the treewidth of (H_1, H_2)-free graphs is unbounded and* Edge Steiner Tree *is* NP*-complete for (H_1, H_2)-free graphs.*

Proof. First suppose that one of the Cases 1–4 holds. Let G be an (H_1, H_2)-free graph. First suppose that $H_1 = K_r$ for some $r \in \{1, 2\}$. Then G has no edges and so has treewidth 0. If $H_1 = K_3$ and $H_2 = K_{1,3}$, then G has maximum degree at most 2, that is, G is the disjoint union of paths and cycles. Hence G has treewidth at most 2. If $H_1 = K_r$ for some $r \geq 3$, and $H_2 = P_3$, then G is the disjoint union of complete graphs, each of size at most $r - 1$. Hence G has treewidth at most $r - 2$. Finally, suppose that $H_1 = K_r$, for some $r \geq 3$, and $H_2 = sP_1$, for some $s \geq 1$. Ramsey's Theorem tells us that for every $r \geq 1$ and $s \geq 1$, there exists a constant $R(r, s)$ such that every graph on at least $R(r, s)$ vertices contains a clique on r vertices or an independent set on s vertices. As G is (K_r, sP_1)-free, we find that the number of vertices of G is bounded by $R(r, s) - 1$.

Hence, G has treewidth at most $R(r,s)-2$. In order to complete the proof of the first statement of the theorem we can now apply Lemma 5 and find that Edge Steiner Tree is polynomial-time solvable for (H_1,H_2)-free graphs.[1]

Now suppose that none of Cases 1–4 apply. We will prove that the treewidth of (H_1,H_2)-free graphs is unbounded and that Edge Steiner Tree is NP-complete for the class of (H_1,H_2)-free graphs.

First suppose that neither H_1 nor H_2 is a complete graph. Then the class of (H_1,H_2)-free graphs contains the class of all complete graphs. As the treewidth of a complete graph K_r is readily seen to be equal to $r-1$, the class of complete graphs, and thus the class of (H_1,H_2)-free graphs, has unbounded treewidth. Moreover, by Lemma 1, Edge Steiner Tree is NP-complete for the class of (H_1,H_2)-free graphs. From now on, assume that $H_1=K_r$ for some $r\geq 1$. As Case 1 does not apply, we find that $r\geq 3$.

Suppose that H_2 contains a cycle C_s as an induced subgraph for some $s\geq 3$. As $H_1=K_r$ for some $r\geq 3$, the class of (H_1,H_2)-free graphs contains the class of (C_3,C_s)-free graphs. As the latter graph class contains the class of $(s+1)$-subdivisions of walls, which have unbounded treewidth due to Lemma 6, the class of (H_1,H_2)-free graphs has unbounded treewidth. Moreover, by Lemma 3, Edge Steiner Tree is NP-complete for the class of (H_1,H_2)-free graphs.

Note that if H_2 contains a cycle as a subgraph, then it also contains a cycle as an induced subgraph. So now suppose that H_2 contains no cycle, that is, H_2 is a forest. First assume that H_2 contains an induced P_1+P_2. Recall that $H_1=K_r$ for some $r\geq 3$. Then the class of (H_1,H_2)-free graphs contains the class of complete bipartite graphs. As this class has unbounded treewidth, the class of (H_1,H_2)-free graphs has unbounded treewidth. Moreover, by Lemma 2, Edge Steiner Tree is NP-complete for the class of (H_1,H_2)-free graphs. From here on we assume that H_2 is a (P_1+P_2)-free forest.

Suppose that H_2 has a vertex of degree at least 3. In other words, as H_2 is a forest, the claw $K_{1,3}$ is an induced subgraph of H_2. Recall that $H_1=K_r$ for some $r\geq 3$. First assume that $r=3$. As Case 2 does not apply, H_2 properly contains an induced $K_{1,3}$. As H_2 is a (P_1+P_2)-free forest, this means that $H_2=K_{1,s}$ for some $s\geq 4$. Then the class of (H_1,H_2)-free graphs contains the class of walls. As the latter class has unbounded treewidth due to Lemma 6, the class of (H_1,H_2)-free graphs has unbounded treewidth. Moreover, by Lemma 3, Edge Steiner Tree is NP-complete for the class of (H_1,H_2)-free graphs. Now assume that $r\geq 4$. Then the class of (H_1,H_2)-free graphs contains the class of net-walls. As the latter graph class has unbounded treewidth due to Lemma 7, the class of (H_1,H_2)-free graphs has unbounded treewidth. Moreover, by Lemma 4, Edge Steiner Tree is NP-complete for the class of (H_1,H_2)-free graphs.

From the above we may assume that H_2 does not contain any vertex of degree 3. This means that H_2 is a linear forest, that is, a disjoint union of paths. As Case 4 does not apply, H_2 has an edge. Every (P_1+P_2)-free linear forest with an edge is either a P_2 or a P_3. However, this is not possible, as Case 1 (with the roles of H_1 and H_2 reversed) and Case 3 do not apply. We conclude that this case cannot happen. □

3. Proof of Theorem 2

In this section we give a proof of our second dichotomy. We state useful past results in Section 3.1 followed by some new results for line graphs and P_4-free graphs in Section 3.2. Then, in Section 3.3, we show how to combine these results to obtain the proof of Theorem 2.

3.1. Known results

The first result we need is due to Brandstädt and Müller. A graph is *chordal bipartite* if it has no induced cycles of length 3 or of length at least 5; that is, a graph is chordal bipartite if it is $(C_3,C_5,C_6,\ldots)$-free.

Theorem 5 *([5]). The unweighted* Vertex Steiner Tree *problem is* NP-*complete for chordal bipartite graphs.*

The second result that we need is due to Farber, Pulleyblank and White. A graph is *split* if its vertex set can be partitioned into a clique and an independent set. It is well known that the class of split graphs coincides with the class of $(2P_2,C_4,C_5)$-free graphs [15].

Theorem 6 *([14]). The unweighted* Vertex Steiner Tree *problem is* NP-*complete for split graphs.*

3.2. New results

We start with the following lemma.

Lemma 8. *The unweighted* Vertex Steiner Tree *problem is* NP-*complete for line graphs.*

[1] Note that the graph under consideration in Cases 1–4 has a very restricted structure: any large connected component (if it exists) is either a path or a cycle. Hence, we can also solve the problem directly instead of applying Lemma 5.

Proof. By Theorem 3, unweighted Edge Steiner Tree is NP-complete. Let (G, U, k) be an instance of this problem. From G we construct a new graph G' by introducing a new vertex v_u for each terminal $u \in U$, which we make only adjacent to u. We let U' consist of all these new vertices. We observe that G' has a Steiner tree T' for U' with at most $k + |U|$ edges if and only if G has a Steiner tree T for U with at most k edges.

We now consider the line graph $L(G')$ of G' with set of terminals $U^* = \{uv_u \mid u \in U\}$; this is a set of edges in G' and a set of vertices in $L(G')$. To complete the proof, we show that G' has a Steiner tree for U' on, say, ℓ edges if and only if $L(G')$ has a Steiner tree for U^* on ℓ vertices. We first note that the edge set E' of a Steiner tree for U' of G' must contain the set U^*. Further, E', considered as a set of vertices of $L(G')$, induces a connected subgraph and has $|E'| = \ell$ vertices. Conversely, if there is a Steiner tree for U^* in $L(G')$ on ℓ vertices, then these vertices, considered as edges in G', form a Steiner tree for U' in G'. □

Recall that a subgraph G' of a graph G is spanning if $V(G') = V(G)$. Let G_1 and G_2 be two graphs. The *join* operation adds an edge between every vertex of G_1 and every vertex of G_2. The *disjoint union* operation takes the disjoint union of G_1 and G_2. A graph G is a *cograph* if G can be generated from K_1 by a sequence of join and disjoint union operations. A graph is a cograph if and only if it is P_4-free [10]. This implies the following well-known lemma.

Lemma 9. *Every connected P_4-free graph on at least two vertices has a spanning complete bipartite subgraph.*

Let G be a graph. For a set S, the graph $G[S] = (S, \{uv \in E(G)\ u, v \in S\})$ denotes the subgraph of G *induced by* S. Note that $G[S]$ can be obtained from G by deleting every vertex of $V(G) \setminus S$. If G has a vertex weighting w, then $w(S) = \sum_{u \in S} w(u)$ denotes the *weight* of S.

Lemma 10. *For every $s \geq 0$,* Vertex Steiner Tree *can be solved in time $O(n^{2s^2-s+5})$ for connected $(sP_1 + P_4)$-free graphs on n vertices.*

Proof. Let $s \geq 0$ be an integer. Let $G = (V, E)$ be a connected $(sP_1 + P_4)$-free graph with a vertex weighting $w : V \to \mathbb{Q}^+$ and set of terminals U. We show how to solve the optimization version of Vertex Steiner Tree on G. Let $R \subseteq V \setminus U$ be such that $G[U \cup R]$ is connected and, subject to this condition, $U \cup R$ has minimum weight $w(U \cup R)$. Thus any spanning tree of $G[U \cup R]$ is an optimal solution. Let us consider the possible size of R.

First suppose that $G[U \cup R]$ is P_4-free. Then, by Lemma 9, $G[U \cup R]$ has a spanning complete bipartite subgraph. That is, there is a bipartition (A, B) of $U \cup R$ such that every vertex in A is joined to every vertex in B. We may assume without loss of generality that $|U| \geq 2$. Then $|U \cup R| \geq 2$, and thus neither A nor B is the empty set. If U intersects both A and B, then $G[U]$ is connected and $|R| = 0$. So let us assume that $U \subseteq A$, and so $R \supseteq B$. Then $R \cap A = \emptyset$ since $G[U \cup B]$ is connected. As we know that every vertex in $A = U$ is joined to every vertex in $B = R$, we find that $|R| = 1$.

Suppose instead that $G[U \cup R]$ contains an induced path P on four vertices. We call the connected components of $G[U]$ *bad* if they do not intersect P or the neighbours of P in G. There are at most $s - 1$ bad components; else, G contains an $sP_1 + P_4$. Let U^* be a subset of U that includes one vertex from each of these bad components. Then each vertex of $G[U \cup R]$ belongs either to U or P or is an internal vertex of a shortest path in $G[U \cup R]$ from P to a vertex of U^*. The number of internal vertices in such a shortest path is at most $2s + 1$; else, the path contains an induced $sP_1 + P_4$. As R is a subset of the union of $V(P)$ and the sets of these internal vertices, we find that $|R| \leq 4 + (2s + 1)(s - 1) = 2s^2 - s + 3$.

So in all cases R contains at most $2s^2 - s + 3$ vertices and our algorithm is just to consider every such set R and check, in each case, whether $G[U \cup R]$ is connected. Our solution is the smallest set found that satisfies the connectivity constraint. As there are $O(n^{2s^2-s+3})$ sets to consider, and checking connectivity takes $O(n^2)$ time, the algorithm requires $O(n^{2s^2-s+5})$ time. □

3.3. The proof

We are now ready to prove our second dichotomy.

Theorem 2 *(restated). Let H be a graph. If H is an induced subgraph of $sP_1 + P_4$ for some $s \geq 0$, then* Vertex Steiner Tree *is polynomial-time solvable for H-free graphs, otherwise even unweighted* Vertex Steiner Tree *is* NP-*complete.*

Proof. If H has a cycle, then we apply Theorem 5 or Theorem 6. Hence, we may assume that H has no cycle, so H is a forest. If H contains a vertex of degree at least 3, then the class of H-free graphs contains the class of claw-free graphs, which in turn contains the class of line graphs. Hence, we can apply Lemma 8. Thus we may assume that H is a linear forest. If H contains a connected component with at least five vertices or two connected components with at least two vertices each, then the class of H-free graphs contains the class of $2P_2$-free graphs. Hence, we can apply Theorem 6. It remains to consider the case where H is an induced subgraph of $sP_1 + P_4$ for some $s \geq 0$, for which we can apply Lemma 10. □

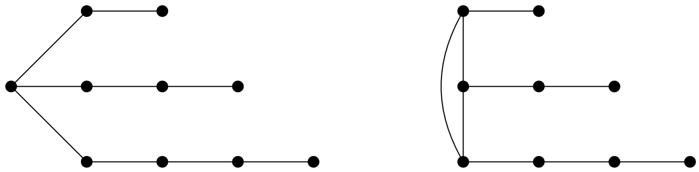

Fig. 5. The graph $S_{2,3,4}$ (left) and the line graph of $S_{2,3,4}$ (right).

4. Conclusions

We presented complexity dichotomies both for Edge Steiner Tree restricted to (H_1, H_2)-free graphs and for Vertex Steiner Tree for H-free graphs. The latter dichotomy also holds for the unweighted variant, in which case the problems Edge Steiner Tree and Vertex Steiner Tree are equivalent due to Observation 1.

In particular, we observed that Edge Steiner Tree can be solved in polynomial time for (H_1, H_2)-free graphs if and only if the class of (H_1, H_2)-free graphs has bounded treewidth (assuming P$\neq$ NP). It is a natural to ask whether Edge Steiner Tree can be solved in polynomial time on a hereditary graph class $\mathcal{G}$ characterized by a *finite* set $\mathcal{F}_{\mathcal{G}}$ of forbidden induced subgraphs if and only if $\mathcal{G}$ has bounded treewidth (assuming P $\neq$ NP). If $\mathcal{F}_{\mathcal{G}}$ does not contain any complete graph or induced subgraph of a complete bipartite graph, then $\mathcal{G}$ contains the class of complete graphs or complete bipartite graphs, respectively. Then $\mathcal{G}$ has unbounded treewidth, and moreover Edge Steiner Tree is NP-complete by Lemma 1 or 2, respectively. The graph $S_{h,i,j}$, for $1 \le h \le i \le j$, is the *subdivided claw*, which is the tree with one vertex x of degree 3 and exactly three leaves, which are at distance h, i and j from x, respectively; see Fig. 5 for an example. Let $\mathcal{S}$ be the class of graphs, every connected component of which is either a subdivided claw or path. Suppose that $\mathcal{F}_{\mathcal{G}}$ does not contain any graph from $\mathcal{S}$ as an induced subgraph. As $\mathcal{F}_{\mathcal{G}}$ is finite, we can take a sufficiently large value of r such that $\mathcal{G}$ contains the class of r-subdivisions of walls. Then $\mathcal{G}$ has unbounded treewidth by Lemma 6 and Edge Steiner Tree is NP-complete for $\mathcal{G}$ by Lemma 3. Let $\mathcal{T}$ be the class of line graphs of graphs of $\mathcal{S}$. By repeating the previous arguments with Lemmas 7 and 4 instead of Lemmas 6 and 3, respectively, we find that $\mathcal{G}$ has unbounded treewidth and that Edge Steiner Tree is NP-complete for $\mathcal{G}$ if $\mathcal{F}_{\mathcal{G}}$ does not contain a graph from $\mathcal{T}$.

To summarize the above, a class of graphs $\mathcal{G}$ with finite $\mathcal{F}_{\mathcal{G}}$ has unbounded treewidth and Edge Steiner Tree is NP-complete for $\mathcal{G}$ if $\mathcal{F}_{\mathcal{G}}$ does not contain any complete graph, or any induced subgraph of a complete bipartite graph, or any graph from $\mathcal{S}$, or any graph from $\mathcal{T}$. In a very recent arXiv paper [20], Lozin and Razgon showed that a class $\mathcal{G}$ with finite $\mathcal{F}_{\mathcal{G}}$ has bounded treewidth if $\mathcal{F}_{\mathcal{G}}$ contains a complete graph, an induced subgraph of a complete bipartite graph, a graph from $\mathcal{S}$ and a graph from $\mathcal{T}$. Recall that Edge Steiner Tree is polynomial-time solvable for graphs of bounded treewidth (Lemma 5). Hence, the result of Lozin and Ragzon implies that Edge Steiner Tree is polynomial-time solvable on a hereditary graph class $\mathcal{G}$ with finite $\mathcal{F}_{\mathcal{G}}$ *if and only if* $\mathcal{G}$ has bounded treewidth (assuming P $\neq$ NP).

The following result shows that the situation changes if $\mathcal{F}_{\mathcal{G}}$ is infinite.

Theorem 7. *There exists a hereditary graph class $\mathcal{G}$ of unbounded treewidth for which* Edge Steiner Tree *can be solved in polynomial time.*

Proof. Let $\mathcal{G}$ consist of graphs G of maximum degree at most 3 such that every path between any two degree-3 vertices in G has at least 2^r vertices, where r is the number of degree-3 vertices in G. As deleting a vertex neither increases the maximum degree of a graph nor shortens any path between a pair of degree-3 vertices, $\mathcal{G}$ is hereditary (note that $\mathcal{G}$ is also closed under taking edge deletion). As $\mathcal{G}$ contains subdivisions of walls of arbitrarily large height, the treewidth of $\mathcal{G}$ is unbounded due to Theorem 4.

We solve Edge Steiner Tree on an instance (G, w, U, k) with $G \in \mathcal{G}$ as follows. If G has at most one vertex of degree 3, then G has treewidth at most 2, so we can apply Lemma 5. Suppose that G has at least two vertices of degree 3. Then we apply the following rules, while possible.

Rule 1. There is a non-terminal x of degree 2. Let xy and xz be its two incident edges. We contract xy and give the new edge weight $w(xy) + w(xz)$. If there was already an edge between y and z, then we remove one with largest weight.

Rule 2. There is a terminal x of degree 2 and its neighbours y and z are also terminals. Assume $w(xy) \le w(xz)$. We observe that there is an optimal solution that includes the edge xy. Hence, we may contract xy and decrease k by $w(xy)$.

Rule 3. There is a vertex x of degree 1. Let y be its neighbour. If x is not a terminal, then remove x. Otherwise, contract xy and decrease k by $w(xy)$.

Let (G', w', U', k') be the resulting instance, which is readily seen to be equivalent to (G, w, U, k). Then G' has r vertices of degree 3 and each vertex of degree at most 2 has a neighbour of degree 3; otherwise, one of Rules 1–3 applies. So, G' has at most $4r$ vertices and thus $O(r)$ edges. It remains to solve Edge Steiner Tree on (G', w', U', k). We do this in $r \cdot 2^{O(r)}$ time by guessing for each edge in G' if it is in the solution and then verifying the resulting candidate solution. Recall that in the problem definition we assume that G is connected. As $r \ge 2$ and G contains at least two vertices of degree 3, this means that $|V(G)| \ge 2^r$. So, the running time is polynomial in $|V(G)|$. $\square$

To obtain a dichotomy for Vertex Steiner Tree and unweighted Vertex Steiner Tree for (H_1, H_2)-free graphs, we need to answer several open problems, including the following two.

Open Problem 1. *Does there exist a pair (H_1, H_2) such that* Vertex Steiner Tree *and unweighted* Vertex Steiner Tree *have different complexities for (H_1, H_2)-free graphs?*

For Open Problem 1, it may be prudent to focus on pairs (H_1, H_2) for which the mim-width of (H_1, H_2)-free graphs is unbounded. This is due to the aforementioned result of Bergougnoux and Kanté [1], who proved that Vertex Steiner Tree is polynomial-time solvable for graph classes of bounded mim-width for which we can compute a branch decomposition of constant mim-width in polynomial time.

Open Problem 2. *For every integer t, determine the complexity of* Vertex Steiner Tree *for $(K_{1,3}, P_t)$-free graphs.*

For Open Problem 2 we note that Vertex Steiner Tree is polynomial-time solvable for P_4-free graphs by Theorem 2. It is known that $(K_{1,3}, P_5)$-free graphs have unbounded mim-width [7]. Hence, the first open case is where $t = 5$. To obtain an answer to Open Problem 2, we need new insights into the structure of $(K_{1,3}, P_t)$-free graphs. These insights may also be useful in obtaining new results for other problems, such as the Graph Colouring problem restricted to $(K_{1,3}, P_t)$-free graphs (see [17,21]).

Declaration of competing interest

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

Acknowledgements

We thank an anonymous reviewer for a number of helpful suggestions.

References

[1] B. Bergougnoux, M.M. Kanté, More applications of the d-neighbor equivalence: connectivity and acyclicity constraints, in: Proc. ESA 2019, LIPIcs, vol. 144, 2019, 17.
[2] M.W. Bern, P.E. Plassmann, The Steiner problem with edge lengths 1 and 2, Inf. Process. Lett. 32 (1989) 171–176.
[3] H.L. Bodlaender, N. Brettell, M. Johnson, G. Paesani, D. Paulusma, Steiner trees for hereditary graph classes, in: Proc. LATIN 2020, LNCS, vol. 12118, 2020, pp. 613–624.
[4] H.L. Bodlaender, M. Cygan, S. Kratsch, J. Nederlof, Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth, Inf. Comput. 243 (2015) 86–111.
[5] A. Brandstädt, H. Müller, The NP-completeness of Steiner tree and dominating set for chordal bipartite graphs, Theor. Comput. Sci. 53 (1987) 257–265.
[6] J. Brault-Baron, F. Capelli, S. Mengel, Understanding model counting for beta-acyclic CNF-formulas, in: Proc. STACS 2015, LIPIcs, vol. 30, 2015, pp. 143–156.
[7] N. Brettell, J. Horsfield, A. Munaro, G. Paesani, D. Paulusma, Bounding the mim-width of hereditary graph classes, in: Proc. IPEC 2020, LIPIcs, vol. 180, 2021, 6.
[8] B.-M. Bui-Xuan, J.A. Telle, M. Vatshelle, Boolean-width of graphs, Theor. Comput. Sci. 412 (2011) 5187–5204.
[9] M. Chimani, P. Mutzel, B. Zey, Improved Steiner tree algorithms for bounded treewidth, J. Discret. Algorithms 16 (2012) 67–78.
[10] D.G. Corneil, H. Lerchs, L. Stewart Burlingham, Complement reducible graphs, Discrete Appl. Math. 3 (1981) 163–174.
[11] J. Chuzhoy, Improved bounds for the flat wall theorem, in: Proc. SODA 2015, 2015, pp. 256–275.
[12] K.K. Dabrowski, M. Johnson, D. Paulusma, Clique-width for hereditary graph classes, in: Proc. BCC 2019, in: London Mathematical Society Lecture Note Series, vol. 456, 2019, pp. 1–56.
[13] D. Du, X. Hu, Steiner Tree Problems in Computer Communication Networks, World Scientific, 2008.
[14] M. Farber, W.R. Pulleyblank, K. White, Steiner trees, connected domination and strongly chordal graphs, Networks 15 (1985) 109–124.
[15] S. Földes, P.L. Hammer, Split graphs, Congr. Numer. 19 (1977) 311–315.
[16] M.R. Garey, D.S. Johnson, The rectilinear Steiner tree problem is NP-complete, SIAM J. Appl. Math. 32 (1977) 826–834.
[17] P.A. Golovach, M. Johnson, D. Paulusma, J. Song, A survey on the computational complexity of colouring graphs with forbidden subgraphs, J. Graph Theory 84 (2017) 331–363.
[18] Ö. Johansson, Clique-decomposition, NLC-decomposition, and modular decomposition – relationships and results for random graphs, Congr. Numer. 132 (1998) 39–60.
[19] R.M. Karp, Reducibility among combinatorial problems, in: Complexity of Computer Computations, 1972, pp. 85–103.
[20] V. Lozin, I. Razgon, Tree-width dichotomy, CoRR, arXiv:2012.01115 [abs], 2020.
[21] B. Martin, D. Paulusma, S. Smith, Colouring H-free graphs of bounded diameter, in: Proc. MFCS 2019, LIPIcs, vol. 138, 2019, pp. 1–14.
[22] S.-I. Oum, Paul Seymour, Approximating clique-width and branch-width, J. Comb. Theory, Ser. B 96 (2006) 514–528.
[23] H.J. Prömel, A. Steger, The Steiner Tree Problem: A Tour Through Graphs, Algorithms, and Complexity, Springer Science & Business Media, 2012.
[24] M. Rao, Clique-width of graphs defined by one-vertex extensions, Discrete Math. 308 (2008) 6157–6165.
[25] P. Renjith, N. Sadagopan, The Steiner tree in $K_{1,r}$-free split graphs - a dichotomy, Discrete Appl. Math. 280 (2020) 246–255.
[26] N. Robertson, P.D. Seymour, Graph minors. V. Excluding a planar graph, J. Comb. Theory, Ser. B 41 (1986) 92–114.
[27] K. Truemper, On the delta-wye reduction for planar graphs, J. Graph Theory 13 (1989) 141–148.