Steiner Trees for Hereditary Graph Classes*

Hans Bodlaender[1][0000−0002−9297−3330], Nick Brettell[2][0000−0002−1136−418X], Matthew Johnson[2][0000−0002−7295−2663], Giacomo Paesani[2][0000−0002−2383−1339], Daniël Paulusma[2][0000−0001−5945−9287], and Erik Jan van Leeuwen[1]

[1] Department of Information and Computing Sciences, Utrecht University, The Netherlands {h.l.bodlaender,e.j.vanleeuwen}@uu.nl
[2] Department of Computer Science, Durham University, UK
{nicholas.j.brettell,matthew.johnson2,giacomo.paesani,daniel.paulusma}@durham.ac.uk

Abstract. 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. 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 $w_E : E \to \mathbb{R}^+$. 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 $w_E(T_U) \leq k$?

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 $w_V : V \to \mathbb{R}^+$. 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.

* Supported by the Leverhulme Trust (RPG-2016-258) and the Royal Society (IES\R1\191223).

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 $w_V(T_U) \leq k$?

Note 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 [7] and Prömel and Steger [14] 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 a problem is *unweighted* if the weighting is constant. It is well known that the unweighted versions of EDGE STEINER TREE and VERTEX STEINER TREE are NP-complete [12,8], and we note that these unweighted problems are polynomially equivalent. We denote instances of the weighted problems by (G, w, U, k) and of the unweighted problems by (G, U, k).

Our Focus We focus on the complexity of EDGE STEINER TREE and VERTEX STEINER TREE for *hereditary* graph classes, i.e., graph classes 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 $\mathcal{H}_{\mathcal{G}} = \{C_3, C_5, \ldots, \}$, 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 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 $\{H_1, \ldots, H_p\}$, a graph is $(H_1, \ldots, H_p)$*-free* if it is H_i-free for every $i \in \{1, \ldots, p\}$. We let K_r and P_r denote the complete graph and path on r vertices. 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, such that for any two distinct vertices u, v, we have $uv \in E$ if and only if $u \in S$ and $v \in T$. We call $K_{1,3}$ the *claw*. In the first dichotomy, the roles of H_1 and H_2 are interchangeable.

Theorem 1. *Let H_1 and H_2 be two graphs.* EDGE STEINER TREE *is polynomial-time solvable for (H_1, H_2)-free graphs if*

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

and otherwise it is NP*-complete.*

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

We make the following observations about these two results:

1. We prove Theorem 1 by pinpointing a strong correspondence to the notion of treewidth. We show, in fact, 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. Although Vertex Steiner Tree is polynomial-time solvable for graph classes of bounded mim-width [1] and thus also for graph classes of bounded treewidth, such a 1-to-1 correspondence does not hold for Vertex Steiner Tree, for treewidth or mim-width. To see this, observe that complete graphs, and hence P_4-free graphs, have unbounded treewidth, whereas cobipartite graphs, and hence $3P_1$-free graphs, have unbounded mim-width. In Section 4 we discuss this connection between Edge Steiner Tree and treewidth further.

2. The restriction of Theorem 1 to monogenic graph classes yields only two (trivial) graphs H, namely $H = P_1$ or $H = P_2$, 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 $H = sP_1 + P_4$ ($s \geq 0$).

3. 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 polynomially equivalent, Theorem 2 is also a classification of the unweighted version of Edge Steiner Tree.

2 The 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 [15] 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 we shall see, this notion will play an important role. We then use these results to prove Theorem 1.

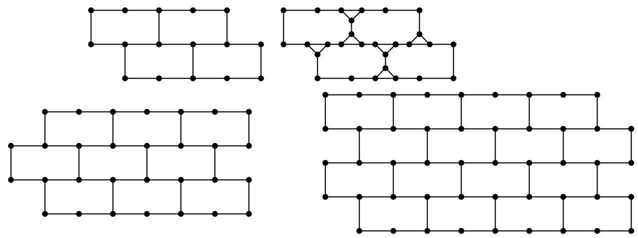

Fig. 1: A wall of height 2, its wye-net transformation, walls of height 3 and 4.

2.1 Preliminaries

The NP-completeness of Edge Steiner Tree on complete graphs follows from the result [12] 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 on a class $\mathcal{C}$ of graphs, then, for every $r \geq 0$, it is so on the class of r-subdivisions of graphs in $\mathcal{C}$.*

We make the following observation (proof omitted).

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

The following follows by inspection of the reduction of Garey and Johnson for Rectilinear Steiner Tree [10]. Let n and m be positive integers. An $n \times m$ grid graph has vertex set $\{v^{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq m\}$ 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 $j < m$).

Theorem 3 ([10]). *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 for three examples of walls of different *heights*. We refer to [6] for a formal definition. Note that walls of height at least 2 have maximum degree 3. 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. 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 after adding the edge a_ib_i for $i = 1, 2, 3$.

We have two results on these classes.

Lemma 3. *For every $r \geq 0$,* 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-hard by Theorem 3. Let (G, U, k) be an instance of unweighted Edge Steiner Tree where G is an $n \times m$ grid graph. 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.

From G, we obtain a graph W as follows. Two vertices of G are exceptional: $v^{n,1}$ is always exceptional, $v^{1,m}$ is exceptional if n is even, and $v^{1,1}$ is exceptional if n is odd. 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 $v^{1,1}_{\downarrow}$ (otherwise). We add an edge from $v^{i,j}_{\downarrow}$ to $v^{i+1,j}_{\uparrow}$, for $1 \leq i \leq n-1$, $1 \leq i \leq m$. For $1 \leq i \leq n$, $1 \leq j \leq m-1$, 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 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 $v^{i,j}_{\uparrow}$ that exists (one or other will not exist if $v^{i,j}$ is exceptional).

We claim that there is a Steiner tree of k edges in G for terminal set U if and only if there is a Steiner tree of weight $k + \delta$ in (W, w') for terminal set U', where $0 \leq \delta < 1$. Indeed, any Steiner tree T in G for terminal set U of k edges corresponds naturally to a Steiner tree T' for U' in (W, w') of weight less than $k + 1$ by adding all new edges to T and letting T' be a spanning tree of the component of the resulting subgraph of W that contains U'. Conversely, any Steiner tree T' for U' in (W, w') of weight $k+\delta$, $0 \leq \delta < 1$, corresponds naturally to a Steiner tree T for U in G of k edges by removing all new edges from T' and letting T be a spanning tree of the resulting subgraph of G. 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 (omitted due to space restrictions).

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

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 non-empty connected subgraph, and, for each edge $vw \in E$, there is at least one node of T that contains v and w.

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 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 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. 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, [5] or, for a faster algorithm [3]).

Lemma 5 ([3,5]). Edge Steiner Tree *can be solved in polynomial time on every graph class of bounded treewidth.*

We also need the well-known Robertson-Seymour Grid-Minor Theorem (also called the Excluded Grid Theorem), which can be formulated for walls.

Theorem 4 ([16]). *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-subdivided walls has unbounded treewidth.*

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

We need the following classification of the boundedness of treewidth for (H_1, H_2)-free graphs (in which we may exchange the roles of H_1 and H_2). Note that this classification coincides with the classification of Theorem 1.

Theorem 5. *Let H_1 and H_2 be two graphs. Then the class of (H_1, H_2)-free graphs has bounded treewidth if and only if*

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

Proof. We first prove that in each of the Cases 1–4, the class of (H_1, H_2)-free graphs has bounded treewidth. 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 $r - 1$. Finally if $H_1 = K_r$, for some $r \geq 3$, and $H_2 = sP_1$, for some $s \geq 1$, then, by Ramsey's Theorem, the number of vertices of G is bounded by some constant $R(r, s)$. Hence G has treewidth at most $R(r, s)$.

We will now show that the class of (H_1, H_2)-free graphs has unbounded treewidth if Cases 1–4 do not apply. 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. 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 $s \geq 1$. 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 subgraphs. As the latter graph class contains the class of $(s+1)$-subdivided walls, which have unbounded treewidth due to Lemma 6, the class of (H_1, H_2)-free graphs has unbounded treewidth.

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. From hereon 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 forest, this means that H_2 contains an induced $P_1 + P_2$, which is not possible. We conclude 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.

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. □

We are now ready to prove Theorem 1.

Theorem 1 (restated). *Let H_1 and H_2 be two graphs.* Edge Steiner Tree *is polynomial-time solvable for (H_1, H_2)-free graphs if and only if*

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

and otherwise it is NP*-complete.*

Proof (Sketch). Let $\mathcal{G}$ denote the class of (H_1, H_2)-free graphs under consideration. If one of Cases 1–4 applies, then $\mathcal{G}$ has bounded treewidth by Theorem 5; we apply Lemma 5. We now show NP-completeness in all remaining cases.

Suppose neither H_1 nor H_2 is a complete graph. Then $\mathcal{G}$ contains all complete graphs, and we apply Lemma 1. 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 $s \geq 1$. Then $\mathcal{G}$ contains all (C_3, C_s)-free graphs. The latter class includes all $(s+1)$-subdivided walls, so we use Lemma 3.

Suppose that H_2 contains no cycle, that is, H_2 is a forest. If H_2 contains an induced $P_1 + P_2$, then $\mathcal{G}$ contains all complete bipartite graphs, and we apply Lemma 2. Now suppose that H_2 has a vertex of degree at least 3. As H is a forest, the claw $K_{1,3}$ is an induced subgraph of H_2. If $r = 3$, then as Case 2 does not apply, H_2 properly contains an induced $K_{1,3}$, which means that H_2 contains an induced $P_1 + P_2$, a contradiction. If $r \geq 4$, then $\mathcal{G}$ contains all net-walls, and we can apply Lemma 4. Now suppose that H_2 does not contain any vertex of degree 3; then H_2 is a linear forest. As Case 4 does not apply, H_2 has an edge. Every $(P_1 + P_2)$-free linear forest with an edge is a P_2 or a P_3. However, this is not possible, as Case 1 and Case 3 do not apply. □

3 The 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 P_4-free graphs in Section 3.2 and 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 6 ([4]). *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 [9].

Theorem 7 ([8]). *The unweighted* VERTEX STEINER TREE *problem is* NP-*complete for split graphs.*

3.2 New Results

We start with the following lemma (proof omitted).

Lemma 8. *The unweighted* VERTEX STEINER TREE *problem is* NP-*complete for line graphs.*

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. 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 $w : V \to \mathbb{R}^+$ 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. 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 (and 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 P and 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. □

We are now ready to prove our second dichotomy.

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

Proof. If H has a cycle, then we apply Theorem 6 or Theorem 7. 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 7. 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. □

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 polynomially equivalent. 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. This correspondence is not true in general.

Theorem 8. *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 decreases the number of vertices on paths between degree-3 vertices, $\mathcal{G}$ is hereditary. As $\mathcal{G}$ contains subdivided 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. Otherwise, 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 $w(xy) \leq w(xz)$. 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. As $r \geq 2$, we have $|V(G)| \geq 2^r$. So, the running time is polynomial in $|V(G)|$. $\square$

As the hereditary graph class $\mathcal{G}$ in Theorem 8 has an infinite family $\mathcal{H}_{\mathcal{G}}$ of forbidden induced subgraphs, we pose the following open problem.

Open Problem 1 *Is* Edge Steiner Tree *polynomial-time solvable for any* finitely defined *hereditary graph class $\mathcal{G}$ if and only if $\mathcal{G}$ has bounded treewidth?*

So far, we have not found any counterexample to Open Problem 1, and to increase our understanding we first aim to consider classes of (H_1, H_2, H_3)-free graphs. The graph $S_{h,i,j}$, for $1 \leq h \leq i \leq j$, is the *subdivided claw*, which is the tree with one vertex x of degree 3 and exactly three leaves, which are of distance h, i and j from x, respectively. Note that $S_{1,1,1} = K_{1,3}$ and that both walls and net-walls may contain arbitrarily large subdivided claws. Note also that complete graphs are C_3-free and complete bipartite graphs are C_4-free. As such we pose the following open problem.

Open Problem 2 *For every subdivided claw S, does the class of (C_3, C_4, S)-free graphs have bounded treewidth?*

We also propose to consider Vertex Steiner Tree and unweighted Vertex Steiner Tree for (H_1, H_2)-free graphs as future research. To obtain a dichotomy, we need to answer several open problems, including the next ones.

Open Problem 3 *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?*

Open Problem 4 *For every integer t, determine the complexity of* Vertex Steiner Tree *for $(K_{1,3}, P_t)$-free graphs.*

To obtain an answer to Open Problem 4, we need new insights into the structure of $(K_{1,3}, P_t)$-free graphs. These insights may also be useful to obtain new results for other problems, such as the Graph Colouring problem restricted to $(K_{1,3}, P_t)$-free graphs (see [11,13]).

References

1. B. Bergougnoux and M. M. Kanté. More applications of the d-neighbor equivalence: Connectivity and acyclicity constraints. *Proc. ESA 2019, LIPIcs*, 144:17:1–17:14, 2019.
2. M. W. Bern and P. E. Plassmann. The Steiner problem with edge lengths 1 and 2. *Information Processing Letters*, 32:171–176, 1989.
3. H. L. Bodlaender, M. Cygan, S. Kratsch, and J. Nederlof. Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth. *Information and Computation*, 243:86–111, 2015.
4. A. Brandstädt and H. Müller. The NP-completeness of Steiner Tree and Dominating Set for chordal bipartite graphs. *Theoretical Computer Science*, 53:257–265, 1987.
5. M. Chimani, P. Mutzel, and B. Zey. Improved Steiner tree algorithms for bounded treewidth. *Journal of Discrete Algorithms*, 16:67–78, 2012.
6. J. Chuzhoy. Improved bounds for the flat wall theorem. *Proc. SODA 2015*, pages 256–275, 2015.
7. D. Du and X. Hu. *Steiner Tree Problems in Computer Communication Networks.* World Scientific, 2008.
8. M. Farber, W. R. Pulleyblank, and K. White. Steiner trees, connected domination and strongly chordal graphs. *Networks*, 15:109–124, 1985.
9. S. Földes and P. L. Hammer. Split graphs. *Congressus Numerantium*, 19:311–315, 1977.
10. M. R. Garey and D. S. Johnson. The rectilinear Steiner tree problem is NP-complete. *SIAM Journal on Applied Mathematics*, 32:826–834, 1977.
11. P. A. Golovach, M. Johnson, D. Paulusma, and J. Song. A survey on the computational complexity of colouring graphs with forbidden subgraphs. *Journal of Graph Theory*, 84:331–363, 2017.
12. R. M. Karp. Reducibility among combinatorial problems. *Complexity of Computer Computations*, pages 85–103, 1972.
13. B. Martin, D. Paulusma, and S. Smith. Colouring H-free graphs of bounded diameter. *Proc. MFCS 2019, LIPIcs*, 138:1–14, 2019.
14. H. J. Prömel and A. Steger. *The Steiner Tree Problem: A Tour through Graphs, Algorithms, and Complexity.* Springer Science & Business Media, 2012.
15. P. Renjith and N. Sadagopan. The Steiner tree in $K_{1,r}$-free split graphs - a dichotomy. *Discrete Applied Mathematics*, to appear.
16. N. Robertson and P. D. Seymour. Graph minors. V. Excluding a planar graph. *Journal of Combinatorial Theory, Series B*, 41:92–114, 1986.