Complexity of the Steiner Network Problem with Respect to the Number of Terminals

Eduard Eiben
Department of Informatics, University of Bergen, Bergen, Norway
eduard.eiben@uib.no

Dušan Knop
Algorithmics and Computational Complexity, Faculty IV, TU Berlin
Department of Theoretical Computer Science, Faculty of Information Technology, Czech Technical University in Prague, Prague, Czech Republic
dusan.knop@tu-berlin.de

Fahad Panolan
Department of Informatics, University of Bergen, Bergen, Norway
fahad.panolan@uib.no

Ondřej Suchý
Department of Theoretical Computer Science, Faculty of Information Technology, Czech Technical University in Prague, Prague, Czech Republic
ondrej.suchy@fit.cvut.cz

Abstract

In the Directed Steiner Network problem we are given an arc-weighted digraph G, a set of terminals $T \subseteq V(G)$ with $|T| = q$, and an (unweighted) directed request graph R with $V(R) = T$. Our task is to output a subgraph $H \subseteq G$ of the minimum cost such that there is a directed path from s to t in H for all $st \in A(R)$.

It is known that the problem can be solved in time $|V(G)|^{O(|A(R)|)}$ [Feldman&Ruhl, SIAM J. Comput. 2006] and cannot be solved in time $|V(G)|^{o(|A(R)|)}$ even if G is planar, unless the Exponential-Time Hypothesis (ETH) fails [Chitnis et al., SODA 2014]. However, the reduction (and other reductions showing hardness of the problem) only shows that the problem cannot be solved in time $|V(G)|^{o(q)}$, unless ETH fails. Therefore, there is a significant gap in the complexity with respect to q in the exponent.

We show that Directed Steiner Network is solvable in time $f(q) \cdot |V(G)|^{O(c_g \cdot q)}$, where c_g is a constant depending solely on the genus of G and f is a computable function. We complement this result by showing that there is no $f(q) \cdot |V(G)|^{o(q^2/\log q)}$ algorithm for any function f for the problem on general graphs, unless ETH fails.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms; Theory of computation → Parameterized complexity and exact algorithms

Keywords and phrases Directed Steiner Network, Planar Graphs, Parameterized Algorithms, Bounded Genus, Exponential Time Hypothesis

Digital Object Identifier 10.4230/LIPIcs.STACS.2019.25

Related Version `https://arxiv.org/abs/1802.08189`

Funding *Eduard Eiben*: Eduard Eiben was supported by Pareto-Optimal Parameterized Algorithms (ERC Starting Grant 715744).
Dušan Knop: Partially supported by DFG under project "MaMu", NI 369/19.
Ondřej Suchý: Supported by grant 17-20065S of the Czech Science Foundation.

36th International Symposium on Theoretical Aspects of Computer Science (STACS 2019).
Editors: Rolf Niedermeier and Christophe Paul; Article No. 25; pp. 25:1–25:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Steiner Tree is one of the most fundamental and well studied problems in combinatorial optimization. The input of Steiner Tree is an edge-weighted undirected graph G and a set $T \subseteq V(G)$ of terminals. Here, the task is to find a least cost connected subgraph H of G containing all the terminals. The problem is known to be NP-complete and, in fact, was one of the 21 NP-complete problems in Karp's original list [29]. The problem is known to be APX-complete, even when the input graph is a complete graph and all edge weights are 1 or 2 [1]. On the other hand, the problem admits a constant factor approximation algorithm and the current best approximation ratio is less than 1.39 [3]. For an overview of the results and applications of Steiner Tree, the reader is referred to monographs [8, 27, 35].

Steiner Tree is well studied in parameterized complexity. The most natural parameter for the problem is the number of terminals q. The first FPT-algorithm for the problem is the $O\big(3^q \cdot n + 2^q \cdot n^2 + n(n \log n + m)\big)$-time algorithm of Dreyfus and Wagner [14] (independently found by Levin [30]) from 1970s; here and on n denotes $|V(G)|$ and m denotes $|E(G)|$. This algorithm, as well as its later improvements [16, 22, 2] subsequently approaching the $O(2^q \operatorname{poly}(n+m))$ running time, uses exponential space. The running time of $O(2^q \operatorname{poly}(n+m))$ is optimal assuming Set Cover Conjecture [9]. There have been many studies for designing algorithms with lower space complexity. Polynomial space FPT-algorithms appeared only recently: First by Nederlof [34] for weights bounded by a constant and later by Fomin et al. [20] for arbitrary weights.

Steiner Tree can be generalized to digraphs. There are many variants of Steiner-type problems on digraphs; the two most natural are Directed Steiner Tree (DST) and Strongly Connected Steiner Subgraph (SCSS). In DST, we are given an arc-weighted directed graph G, a set $T \subseteq V(G)$ of q terminals, and a root vertex $r \in V(G)$. Our task is to find a least cost subgraph H of G such that for every $t \in T$, t is reachable from r in H. In SCSS, the input is an arc-weighted directed graph G and a set $T \subseteq V(G)$ of terminals. The task is to find a least cost subgraph H of G such that for every $s, t \in T$, there are directed paths from s to t and from t to s in H. That is, H is a least cost strongly connected subgraph containing all the terminals. A common generalization of DST and SCSS is Directed Steiner Network (DSN). In DSN, we are given an arc-weighted digraph G, a set $T \subseteq V(G)$ of q terminals, and a digraph R on T. The task is to find a least cost subgraph H of G which realizes all paths prescribed by the arcs of R. That is, for every arc $st \in A(R)$, there is a directed path from s to t in H. Observe that, in DSN, request graphs R and R' yield the same set of solutions if their transitive closures are the same. DST is a special case of DSN where R is a single out-tree on $T \cup \{r\}$ with r being the root and T being the set of leaves. Similarly, SCSS is a special case of DSN where R is a single directed cycle on T.

Existence of an α-approximation algorithm for DST implies a 2α-approximation algorithm for SCSS because of the following simple observation. The union of an in-tree and an out-tree from one fixed terminal in T yields a strongly connected subgraph containing T. The best known approximation ratio in polynomial time for DST and SCSS is $O(q^\varepsilon)$ for any $\varepsilon > 0$ [4]. The same paper also yields an $O(\log^2 q)$-approximation algorithm in quasi-polynomial time. A result of Halperin and Krauthgamer [26] implies that DST and SCSS have no $O(\log^{2-\varepsilon} n)$-approximation for any $\varepsilon > 0$, unless NP has quasi-polynomial time Las Vegas algorithms. The best known approximation algorithm for DSN is by Chekuri et al. [5] with an approximation factor of $O(|A(R)|^{1/2+\epsilon})$ for any $\epsilon > 0$. On the other hand, DSN cannot be approximated to within a factor of $O(2^{\log^{1-\epsilon} n})$ for any fixed $\epsilon > 0$, unless NP $\subseteq$ TIME$(2^{\operatorname{polylog}(n)})$ [13].

Recently Dinur and Manurangsi [12] showed that, under ETH, no polynomial time algorithm and, under Gap-ETH, even no algorithm parameterized by q can approximate DSN to within a factor of $O(|A(R)|^{1/4-o(1)})$.

Using essentially the same techniques as for Steiner Tree [14], one can show that there is an $O\big(3^q \cdot n + 2^q \cdot n^2 + n(n \log n + m)\big)$ time algorithm for DST. On the other hand, Guo et al. [25] showed that SCSS parameterized by q is W[1]-hard. That is, there is no $f(q) \cdot n^{O(1)}$ time algorithm for SCSS for any function f, unless FPT=W[1]. Later a stronger lower bound has been shown by Chitnis et al. [7]. They showed that, in fact, there is no $f(q)n^{o(q/\log q)}$ algorithm for SCSS for any function f, unless Exponential Time Hypothesis (ETH) of Impagliazzo and Paturi [28] fails. This stimulated the research on DSN for restricted classes of request graphs [36, 18] and host graphs [6].

As DSN is a generalization of SCSS, DSN is also W[1]-hard parameterized by q. On the positive side, Feldman and Ruhl [17] showed that DSN can be solved in $n^{O(|A(R)|)}$ time. An independent algorithm with a similar running time also follows from the classification work of Feldmann and Marx [18]. Complementing these results Chitnis et al. [7] showed that DSN cannot be solved in $f(q)n^{o(|A(R)|)}$ time for any function f, even when restricting the host graph G to be planar and all arc weights equal to 1, unless ETH fails. In this reduction (as well as in the reduction given for SCSS), the number of arcs of the request graph $|A(R)|$ is only linear in the number of terminals q. Hence, viewed in terms of the number of terminals, this lower bound implies that there is no $f(q)n^{o(q)}$ time algorithm for any function f, unless ETH fails. But both the known algorithms have running time $n^{\Theta(q^2)}$ in the worst case, leaving a significant gap between the upper and the lower bound for DSN. In this work we contribute to fill this gap.

1.1 Our Results

▶ **Theorem 1.1.** *There is an algorithm which solves any instance* (G, R) *of DSN in time* $2^{c_g q^2 \log(c_g q)} \cdot n^{O(c_g \cdot q)}$, *where* g *is the Euler genus of the graph* G *and* $c_g = 20^{8g+12} g$.

The main idea behind the algorithm is as follows. Let H be a least cost subgraph of G which realizes all paths prescribed by the arcs of R (call it *an optimum solution*). By the result of Feldmann and Marx [18], if the treewidth[1] of H is ω, then there is an algorithm for solving DSN running in time $2^{O(q \cdot \omega \log \omega)} \cdot n^{O(\omega)}$.[2] Towards proving Theorem 1.1 we construct a graph H' from H such that

- the genus of H' is at most g (recall that g is the genus of the input graph G),
- H' and H have the same grid minors and hence $\mathrm{tw}(H) \leq 20^{4\cdot(2g+3)}\,\mathrm{tw}(H')$, and
- the diameter of H' is $O(q)$.

Finally, since H' has genus g and diameter $O(q)$, it follows from a result of Eppstein [15] that $\mathrm{tw}(H') = O(g \cdot q)$. We conclude that H has treewidth $O(c_g \cdot q)$ and our result follows using the algorithm of Feldmann and Marx [18].

We complement the above positive result by the following negative one for general graphs.

▶ **Theorem 1.2.** *There is no* $f(q) \cdot n^{o(q^2/\log q)}$ *time algorithm for DSN on general graphs for any function* f, *unless ETH fails.*

Towards this result, we give a reduction from Partitioned Subgraph Isomorphism (PSI).

[1] Since H is a directed graph, we have to clarify that by the treewidth of a directed graph we mean the treewidth of the underlying undirected graph of H (that is, in a graph on the same vertex set that contains an edge $\{u, v\}$ if and only if H contained an arc (u, v)).

[2] The exact running time bound is more complicated, see Proposition 2.1 for exact statement.

2 Preliminaries

For a positive integer η, we use $[\eta]$ to denote the set $\{1, \ldots, \eta\}$. We consider simple directed graphs and use mostly standard notation that can be found for example in the textbook by Diestel [11]. Let G be a directed graph and let $V(G)$ and $A(G)$ denote its vertex set and arc set, respectively. For vertices $u, v \in V(G)$ the arc from u to v is denoted by uv or (u, v). A *walk* $P = (p_0, \ldots, p_\ell)$ of length ℓ in G is a tuple of vertices, that is, $p_i \in V(G)$ for all $0 \leq i \leq \ell$, such that $p_i p_{i+1} \in A(G)$ for all $0 \leq i < \ell$. A *directed path* $P = (p_0, \ldots, p_\ell)$ in G is a walk of length ℓ with all vertices distinct, that is $p_i \neq p_j$ for all $0 \leq i < j \leq \ell$. We let $V(P) = \{p_0, \ldots, p_\ell\}$. We say that the path P is from p_0 to p_ℓ; we call p_0 and p_ℓ the *endpoints* of P while the other vertices of P are called *internal* (we denote the set of all internal vertices of P by $\mathring{P}$). Path P is between u and v if it is either from u to v or from v to u. Let W be a set of vertices, we say that a path Q is a *W-avoiding path* if $\mathring{Q} \cap W = \emptyset$; if P is a path we say that Q is *P-avoiding path* if it is a $V(P)$-avoiding path. Let P be a walk from u to v and let Q be a walk from v to w. By $P \circ Q$ we denote the *concatenation* of P and Q, that is, the walk from u to w that follows P from u to v and then follows Q from v to w. Let $P = (p_0, \ldots, p_\ell)$ be a directed path and $u, v \in V(P)$. We write $u \leq_P v$ if u is before v on P, in other words, $u = p_i$ and $v = p_j$ such that $i \leq j$. Furthermore, for $i \leq j$ the *subpath of P between p_i and p_j*, denoted $p_i[P]p_j$, is the path $(p_i, \ldots, p_j)$.

For a vertex $v \in V(G)$ its *in-degree* is defined as $\deg^-_G(v) = |\{u \in V \mid uv \in A(G)\}|$. The *out-degree* of v is $\deg^+_G(v) = |\{u \in V \mid vu \in A(G)\}|$. Finally, the *total degree* of v is $\deg_G(v) = \deg^-_G(v) + \deg^+_G(v)$. If the graph G is clear from the context we drop the subscript G. We use $\mathrm{sym}(G)$ to denote the underlying undirected graph of a directed graph G. To *subdivide an arc* $e \in A(G)$ is to delete $e = uv$, add a new vertex w, and add the arcs uw, wv. We say that H is a *subdivision* of G if it can be obtained by repeated subdivision of arcs of G, that is, there exist graphs $G = G_0, \ldots, G_\eta = H$ such that G_{i+1} is the result of arc subdivision in G_i.

We consider the following problem:

Directed Steiner Network (DSN)	
Input:	An arc-weighted directed graph G and an (unweighted) directed graph R such that $V(R) \subseteq V(G)$.
Question:	Find a minimum-cost subgraph H of G in which there is a path from s to t for every $st \in A(R)$.

The problem is also called Directed Steiner Forest or Point-to-Point Connection. We only consider positive weights on arcs, since it is possible to include all non-positive weight arcs into the solution. We call a subgraph H of G *a solution* to the instance (G, R) of DSN if H contains a path from s to t for every $st \in A(R)$. Moreover, we say that H is an *inclusion-minimal* solution to R, if H is a solution for some instance (G, R), but for every $e \in A(H)$, $H - e$ is not. Note that an optimum solution (one with the least sum of weights) is necessarily inclusion-minimal, as we assume positive weights.

► **Proposition 2.1** (Feldmann and Marx [18, Theorem 5] (see also [19])). *Let an instance of DSN be given by a graph G with n vertices and a pattern R on q terminals with vertex cover number τ. If the optimum solution to R in G has treewidth ω, then the optimum can be computed in time $2^{O(q+\tau\omega\log\omega)}n^{O(\omega)}$.*

► **Proposition 2.2** (Demaine, Hajiaghayi, and Kawarabayashi [10]). *Suppose G is a graph with no $K_{3,k}$-minor. If the treewidth of G is at least $20^{4k}r$, then G has an $r \times r$ grid minor.*

For the rest of the paper, by the genus of a graph we always mean *Euler genus*; that is the minimum integer g such that the graph can be drawn without crossing itself on a sphere with g cross-caps or with $g/2$ handles. For a more detailed treatment of topological graph theory the reader is referred to [33] or [24].

▶ **Proposition 2.3** (Ringel, see [33, Theorem 4.4.7]). *If G has Euler genus at most g, then G does not contain $K_{3,2g+3}$ as a minor.*

▶ **Proposition 2.4** (Eppstein [15, Theorem 2]). *Let G be a graph of Euler genus*[3] *g and diameter D. Then G has treewidth $O(gD)$.*

t-Boundaried Graphs and Gluing. A *t-boundaried graph* is a graph G and a set $B \subseteq V(G)$ of size at most t with each vertex $v \in B$ having a label $G(v) \in \{1, \dots, t\}$. Each vertex in B has a unique label. We refer to B as the boundary of G. For a t-boundaried graph G the function $\delta(G)$ returns the boundary of G. Two t-boundaried graphs G_1 and G_2 can be *glued* together to form a graph $G = G_1 \oplus G_2$. The gluing operation takes the disjoint union of G_1 and G_2 and identifies the vertices of $\delta(G_1)$ and $\delta(G_2)$ with the same label.

A t-boundaried graph H is a minor of a t-boundaried graph G if (a t-boundaried graph isomorphic to) H can be obtained from G by deleting vertices or edges or contracting edges, but never contracting edges with both endpoints being boundary vertices[4]. For more details see e.g. [21].

Monadic Second Order Logic. The syntax of Monadic second order logic (MSO) includes the logical connectives $\vee, \wedge, \neg, \Rightarrow, \Leftrightarrow$, variables for vertices, edges, sets of vertices and sets of edges, the quantifiers $\forall, \exists$ that can be applied to these variables, and the following five binary relations:

1. $u \in U$ where u is a vertex variable and U is a vertex set variable;
2. $d \in D$ where d is an edge variable and D is an edge set variable;
3. $\mathrm{inc}(d, u)$, where d is an edge variable, u is a vertex variable; and the interpretation is that the edge d is incident on the vertex u;
4. $\mathrm{adj}(u, v)$, where u and v are vertex variables, and the interpretation is that u and v are adjacent;
5. equality of variables representing vertices, edges, set of vertices and set of edges.

Many common graph-theoretic notions such as vertex degree, connectivity, planarity, outerplanarity, being acyclic, and so on, can be expressed in MSO, as can be seen from introductory expositions [31].

3 Solving DSN on a Fixed Surface

Fix an instance (G, R) of DSN. Let the genus of G be a fixed constant g and let H be an inclusion-minimal solution to (G, R). Note that, since H is a subgraph of G, the genus of H is at most g.

The goal of this section is to show the following theorem.

---

[3] The original paper of Eppstein states genus instead of Euler genus; however, the proof works for both orientable and non-orientable genus and hence also for Euler genus.

[4] Note that these operations preserve the labeling of the boundary vertices.

▶ **Theorem 3.1.** *Let g be a fixed constant. If (G, R) is an instance of DSN such that the genus of G is at most g and H is an inclusion-minimal solution to (G, R), then the treewidth of H is $O\big(20^{4(2g+3)}g \cdot q\big)$.*

With this theorem at hand Theorem 1.1 follows from Proposition 2.1. Note that we can treat every connected component of H separately. More precisely, for each connected component H_C of H, we apply the rest of the proof to H_C and $R[T \cap V(H_C)]$. Hence, we assume that H is connected.

Reversing Arcs – Symmetry. Let $\overleftarrow{G}$, $\overleftarrow{H}$, and $\overleftarrow{R}$ be the directed graphs we obtain from G, H, and R, respectively, by reversing all the arcs. That is, for example, $\overleftarrow{G}$ contains an arc uv if and only if G contains the arc vu. Note that there is a one-to-one correspondence between an s-t path in H and a t-s path in $\overleftarrow{H}$. Hence, if H is an optimum solution to the instance (G, R), then $\overleftarrow{H}$ is an optimum solution to the instance $\big(\overleftarrow{G}, \overleftarrow{R}\big)$. The importance of $\overleftarrow{G}, \overleftarrow{H}, \overleftarrow{R}$ is that every lemma holds in both H, R and $\overleftarrow{H}, \overleftarrow{R}$. In this way we obtain symmetric results without reproving everything twice.

▶ **Lemma 3.2.** *Let (G, R) be an instance of DSN, H be an inclusion-minimal solution to (G, R), and let H be connected. Let R' be a directed graph with vertex set T and for every $s, t \in T$ with $s \neq t$ satisfying $st \in A(R')$ if and only if there is a T-avoiding s-t path in H. Then the following holds:*

1. *H is an inclusion-minimal solution to R' and*
2. *$\mathrm{sym}(R')$ is connected.*

Proof. Assume for the contradiction that $\mathrm{sym}(R')$ is not connected and let R_1 be a connected component of R'. Note that since H is inclusion-minimal every vertex of H lies on some s-t path with $s, t \in T$ and $st \in A(R')$. Now let V_1 be the set of vertices that lie on some s-t path in H for $s, t \in V(R_1)$ and $V_2 = V(H) \setminus V_1$. Clearly, $T \setminus V(R_1) \subseteq V_2$, hence V_2 is not empty. Otherwise, by the definition of R', R' would contain an arc between a vertex in $V(R_1)$ and $V(R') \setminus V(R_1)$. Moreover, every vertex in V_2 lies on some terminal-to-terminal path for two terminals in $T \setminus V(R_1)$. Now let $u \in V_1$ and $v \in V_2$. Clearly, u lies on some s_1-t_1 path between two terminals in R_1 and v lies on a s_2-t_2 path between two terminals in $T \setminus V(R_1)$. Since R' does not contain arcs s_1t_2 nor s_2t_1, it follows that there is no arc between u and v. Since this is true for any two vertices $u \in V_1$ and $v \in V_2$ it follows that $H[V_1]$ is a connected component of H, which contradicts the assumption that H is connected. ◀

From now on we replace R with R'.

▶ **Definition 3.3.** *Let H_1, H_2 be two directed graphs. We say that the pair (H_1, H_2) is a c-*admissible pair *if the genus of H_2 is at most the genus of H_1 and $\mathrm{tw}(H_1) \leq c \cdot \mathrm{tw}(H_2)$.*

Overview of the Proof of Theorem 3.1. We transform the solution graph H into a graph H' containing all terminals and preserving all terminal-to-terminal connections such that (H, H') is an c-admissible pair for some constant c and H' has bounded diameter (and thus by Proposition 2.4 has bounded treewidth). We do this by exploiting that a terminal-to-terminal path in H contains only $O(q)$, so called, important and marked vertices. Furthermore, a subpart of the solution "between" two consecutive marked or important vertices has constant treewidth and contains few vertices with arcs to the rest of the solution H. This allows us to reduce this part of the solution to constant size while preserving genus and all terminal-to-terminal connections. Thus, obtaining the graph H' of bounded diameter.

The following lemma shows that we can assume that each non-terminal vertex in H has at least 3 neighbors.

▶ **Lemma 3.4.** *Let (G, R) be an instance of DSN, H be an inclusion-minimal solution to (G, R), and let H be connected. There exists a directed graph $H_{\geq 3}$ such that $H_{\geq 3}$ is an inclusion-minimal solution to R, $H_{\geq 3}$ is connected, every non-terminal vertex in $H_{\geq 3}$ has at least three neighbors, and $(H, H_{\geq 3})$ is a 1-admissible pair. Moreover, for any $s, t \in T$, there is a T-avoiding s-t path in H if and only if there is one in $H_{\geq 3}$.*

Proof Sketch. We exhaustively repeat the following. Let v be a non-terminal vertex and suppose u, w are the only two neighbors of v. Note that v cannot have only one neighbor, since H is an inclusion-minimal solution. We delete v from H and add an arc uw if both uv and vw were in H, similarly for an arc wu. Denote the resulting graph $H_{\geq 3}$. ◀

3.1 Important and Marked Vertices

For a fixed T-avoiding directed path P in H between two terminals s and t, we say that a vertex $u \in V(P)$ is *important with respect to* P if there is a P-avoiding directed path from some terminal not on P to u or from u to some terminal not on P. Let I_P denote the set of all important vertices with respect to P. Let I be the union of important vertices over all T-avoiding paths in H between terminals.

Let $s, t \in T$ and $P = (s = p_1, \ldots, p_r = t)$ be fixed for the rest of this subsection.

▶ **Lemma 3.5.** *Let (G, R) be an instance of DSN and H be an inclusion-minimal solution to (G, R). Let $P = (s = p_1, \ldots, p_r = t)$ be a T-avoiding directed path between $s, t \in T$. There are at most $2q - 2$ important vertices on P. Moreover, there exists a function $g_P \colon I_P \to T$ with $\left|g_P^{-1}(x)\right| \leq 2$ for every $x \in T$ such that for every $v \in I_P$ there is either v-$g(v)$ or $g(v)$-v directed $(V(P) \cup T)$-avoiding path.*

Proof. We bound the number of important vertices by inspecting the interaction between the path P and other paths in the solution H. In order to do this, we construct a partial labeling $\mathcal{L} \colon V(P) \to 2^{(T \times \{\leftarrow, \rightarrow\})}$ as follows. For a vertex $v \in V(P)$ we have $(x, \leftarrow) \in \mathcal{L}(v)$ if there is a directed P-avoiding path from a terminal x to v in H and v is the closest to s among all such vertices of P. Similarly, we have $(x, \rightarrow) \in \mathcal{L}(v)$ if there is a directed P-avoiding path from v to a terminal x in H and v is the closest to t among all such vertices of P.

▷ Claim 3.6 ($\star$[5]). Every important vertex received some label.

It follows from the above claim that the number of important vertices is bounded by the possible number of labels which is $2q - 2$. This is because by the definition of the labeling every label in $T \times \{\leftarrow, \rightarrow\}$ is used at most once and $(s, \leftarrow)$ and $(t, \rightarrow)$ labels are never assigned to any vertex of P (as they would be assigned to s and t, respectively).

As to the moreover part, it follows from the labeling procedure that if $(\leftarrow, x) \in \mathcal{L}(v)$, then there is a P-avoiding path Q in H from x to v. If this path contains another terminal, then let y be the terminal closest to v on Q. We claim that $(\leftarrow, y) \in \mathcal{L}(v)$ as well. If not, then there would be another vertex v' on P with this label with $v' <_P v$ and a P-avoiding path Q' from y to v'. But then $x[Q]y \circ y[Q']v'$ is a walk that can be shortened to a P-avoiding path from x to v', contradicting $(x, \leftarrow) \in \mathcal{L}(v)$. Hence, each important vertex has a $(V(P) \cup T)$-avoiding path to or from some terminal, such that it has a label of that terminal. To prove the moreover part it remains to set $g_P(v)$ to any such terminal. ◀

[5] Proofs of claims and lemmata marked with $(\star)$ were deferred to the full version of the paper.

▶ **Lemma 3.7** ($\star$). *Let (G, R) be an instance of DSN and H be an inclusion-minimal solution to (G, R). Let $P = (s = p_1, \ldots, p_r = t)$ be a T-avoiding directed path between $s, t \in T$. If v is a vertex in $V(P) \setminus I_P$, then its out-degree is at most 2. Moreover, if u is its out-neighbor not on P, then there is a P-avoiding path from u to some vertex $v' \in V(P)$ with $v' <_P v$.*

The following expresses that in order to bound the diameter of H' it is enough to bound the length of the path P linearly in $|I_P|$.

▶ **Lemma 3.8.** *Let (G, R) be an instance of DSN, H be an inclusion-minimal solution to (G, R), and let H be connected. Moreover, assume that for every $s, t \in T$ with $s \neq t$ that $st \in A(R)$ if and only if there is a T-avoiding s-t path in H. If for every $\bar{s}\bar{t} \in A(R)$ there is a T-avoiding path $\tilde{P}$ in H of length at most $c \cdot |I_{\tilde{P}}|$, for some constant c, then the distance between any two terminal vertices in the underlying undirected graph* $\mathrm{sym}(H)$ *of H is at most $8cq$.*

Proof. By assumption and Lemma 3.2 both H and R are connected. Let $t_1, t_2 \in T$ be two arbitrary terminal vertices and let $Q = (t_1 = t^1, t^2, \ldots, t^\ell = t_2)$ be a shortest path from t_1 to t_2 in $\mathrm{sym}(R)$. Now let $\mathcal{Q} = (Q_1, \ldots, Q_{\ell-1})$ be a realization of the path Q in H, that is, Q_i is a directed T-avoiding path between t^i and t^{i+1} of length at most $c \cdot |I_{Q_i}|$ in H for every $1 \leq i \leq \ell - 1$. Note that it does not matter whether Q_i is a directed path from t^i to t^{i+1} or vice versa.

For $1 \leq i \leq \ell - 1$ let g_i be the function g_{Q_i} for the path Q_i from Lemma 3.5. Let $v \in I_{Q_i}$ be an important vertex on Q_i. From Lemma 3.5 it follows that there is a $(V(Q_i) \cup T)$-avoiding directed path either from v to $g_i(v)$ or from $g_i(v)$ to v. Moreover, since Q_i is T-avoiding, there are two T-avoiding directed paths in H either one from t^i to v and the other from v to t^{i+1} or one from t^{i+1} to v and the other from v to t^i. Therefore, it follows that if a terminal t' is in $g_i(I_{Q_i})$, then there is a T-avoiding directed path either between t' and t^i or between t' and t^{i+1} in H and consequently, by our assumptions on R, there is an arc between t' and either t^i or t^{i+1} in R.

Now, for a terminal t', let $1 \leq i < j \leq \ell - 1$ be such that $t' \in \big(g_i(I_{Q_i}) \cap g_j(I_{Q_j})\big)$. Then we claim that $j - i \leq 3$. From the argument above, it follows that there is an edge between t' and t^i or t^{i+1} and between t' and t^j or t^{j+1} in $\mathrm{sym}(R)$. However, if $j - i \geq 4$, then we can obtain a shorter path than Q in $\mathrm{sym}(R)$ from t_1 to t_2 by going along Q from t_1 to t^i or to t^{i+1}, then using the aforementioned edges to t' and from t' to t^j or t^{j+1} and continuing on Q. This is a contradiction with the choice of Q. Therefore, for each terminal t' there are at most 4 paths $\bar{Q} \in \mathcal{Q}$ such that $t' \in g_{\bar{Q}}(I_{\bar{Q}})$. Since for each path $\bar{Q}$ and terminal t', it holds that $\left|g_{\bar{Q}}^{-1}(t')\right| \leq 2$, it follows that $\sum_{i=1}^{\ell-1} |I_{Q_i}| \leq 2 \cdot 4 \cdot q$. Therefore, the distance between t_1 and t_2 is at most $\sum_{i=1}^{\ell-1} |Q_i| \leq \sum_{i=1}^{\ell-1} c \cdot |I_{Q_i}| \leq 8cq$ and the lemma follows. ◀

▶ **Lemma 3.9.** *Let (G, R) be an instance of DSN and H be an inclusion-minimal solution to (G, R). Let $P = (s = p_1, \ldots, p_r = t)$ be a T-avoiding directed path between $s, t \in T$. Let p_i, p_j, p_k be three vertices on P such that*

(1) *$i < j < k$,*

(2) *there is a path Q from p_k to p_i that avoids p_j, and*

(3) *every directed path P' from some terminal s' to p_j in H intersects P in a vertex p_ℓ such that $p_\ell \neq p_j$ and $\ell \leq k$.*

Then p_j has no in-neighbor other than p_{j-1} in H.

Proof. Refer to Fig. 1. Let $u \neq p_{j-1}$ be an in-neighbor of p_j. Let $s't'$ be an arc in R such that the arc up_j is on a path P' from s' to t' in H. We show that there is a directed path from s' to t' in $H - up_j$. By our assumption, it follows that $s'[P']p_j$ intersects P in a vertex

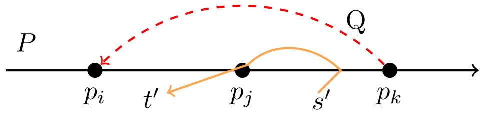

Figure 1 The three vertices p_i, p_j, p_k on a directed path P as in Lemma 3.9. The orange (light gray) path cannot exist as it is rerouted via p_k and Q (dashed); contradicting the minimality of the solution.

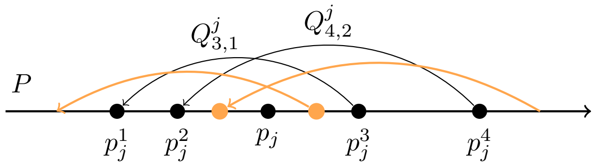

Figure 2 The four vertices on P for the vertex p_j. By the choice of p^1_j and p^4_j, the orange (light gray) paths cannot exist.

p_ℓ such that $\ell < k$. Therefore, the walk $s'[P']p_\ell \circ p_\ell[P]p_k \circ p_k[Q]p_i \circ p_i[P]p_j \circ p_j[P']t'$ induces a directed path from s' to t' in $H - up_j$. Since this is true for every pair of terminals s', t' with an s'-t' path in H, it contradicts the inclusion-minimality of H and hence the only in-neighbor of p_j is p_{j-1}. ◀

For a vertex $p_j \in V(P)$ let $p^1_j, p^2_j, p^3_j, p^4_j$ denote the following four vertices (see Fig. 2):

- p^1_j is the $\le_P$-minimal vertex on P such that there is a P-avoiding path from a vertex p_x, with $x \ge j$ to p^1_j,
- p^3_j is a vertex such that $p_j \le_P p^3_j$ and p^3_j is the first vertex of some P-avoiding path to p^1_j,
- p^4_j is the $\le_P$-maximal vertex on P such that there is a P-avoiding path from p^4_j, to some vertex p_y with $y \le j$, and
- p^2_j is a vertex such that $p^2_j \le_P p_j$ and p^2_j is the last vertex of some P-avoiding path from p^4_j.

Furthermore, let us denote $Q^j_{3,1}$ and $Q^j_{4,2}$ the P-avoiding paths from p^3_j to p^1_j and from p^4_j to p^2_j, respectively, and let $Q^j_{4,1}$ denote the path $Q^j_{4,2} \circ p^2_j[P]p^3_j \circ Q^j_{3,1}$.

▶ **Lemma 3.10** (⋆). *Let (G, R) be an instance of DSN and H be an inclusion-minimal solution to (G, R) such that every non-terminal vertex in H has at least three neighbors. Let $P = (s = p_1, \dots, p_r = t)$ be a T-avoiding directed path between $s, t \in T$. For every p_j there are at most two vertices in $V(P) \setminus (I_P \cup \{p^2_j, p^3_j\})$ between p^1_j and p^4_j.*

For the rest of this section, let us define the set $Q_P = \{p^1_j, p^2_j, p^3_j, p^4_j \mid p_j \in I_P\}$. Clearly, $|Q_P| \le 4|I_P|$. We will call the set Q_P the *set of marked vertices for P*. Note that the same vertex in Q_P may be marked for different reasons at the same time. That is, for example, the same vertex can be denoted p^1_j, because it is the first vertex for the important vertex p_j and at the same time it can be denoted p^3_k, because it is also the third marked vertex for the important vertex p_k with respect to P.

3.2 Ladders

In this subsection we define ladder graphs. These graphs play crucial a role as we will be able to show that if there is a T-avoiding s-t path for $st \in A(R)$ that is "long", then in H there is a "large" ladder (Lemma 3.13). Moreover, it is possible to replace such a ladder with one having constant size while preserving all connections and inclusion-minimality (Lemma 3.14).

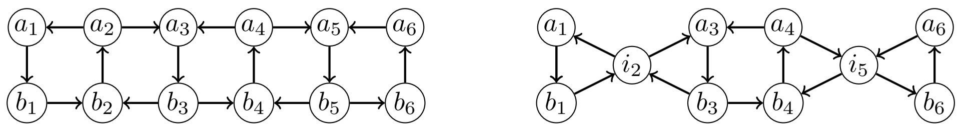

Figure 3 Example ladder graphs. The ladder $G_6 = G_{6,\emptyset}$ on the left and $G_{6,\{2,5\}}$ on the right.

▶ **Definition 3.11** (Class of Ladders). *Let η be a positive integer and $I \subseteq [\eta]$. We define the directed graph G_η and the directed graph $G_{\eta,I}$ as follows (see Fig. 3). The vertex set $V(G_\eta)$ is the set $\{a_i, b_i \mid i \in [\eta]\}$ and the arc set $A(G_\eta)$ is the set*

$$\{a_{2i+1}b_{2i+1} \mid 0 \le i < \eta/2\} \cup \{b_{2i}a_{2i} \mid 1 \le i \le \eta/2\} \cup \{a_{2i}a_{2i-1} \mid 1 \le i \le \eta/2\} \cup \{a_{2i}a_{2i+1} \mid 1 \le i < \eta/2\} \cup \{b_{2i+1}b_{2i} \mid 1 \le i < \eta/2\} \cup \{b_{2i-1}b_{2i} \mid 1 \le i \le \eta/2\}.$$

The graph $G_{\eta,I}$ is the graph G_η where we identify the vertices a_i and b_i whenever $i \in I$ (i.e., G_η and $G_{\eta,\emptyset}$ is the same graph). We emphasize that we suppress any loops in $G_{\eta,I}$. We say that η is the length *of the ladder $G_{\eta,I}$.*

▶ **Lemma 3.12** ($\star$). *Given a positive integer η and $I \subseteq [\eta]$, the ladder $G_{\eta,I}$ is a union of two paths P_1 from a_1 to a_η and P_2 from b_η to b_1 if η is even or paths P_1 from a_1 to b_η and P_2 from a_η to b_1, if η is odd. Moreover, $G_{\eta,I}$ is an inclusion-minimal strongly connected graph connecting the set of terminals $\{a_1, b_1, a_\eta, b_\eta\}$.*

3.3 Finishing the Proof

Let again P be a T-avoiding directed path in H between two terminals s and t. In the following technical lemma we show that if the distance on P between any two consecutive vertices $p_i, p_j \in Q_P \cup I_P$ with $i < j$ is at least 5, then $p_i = p_k^4$ and $p_j = p_\ell^1$ where $p_k, p_\ell \in I_P$ and $k < \ell$. Moreover, there exists a path from p_j to p_i in H and between p_i and p_j there is a ladder with a constant-sized boundary.

▶ **Lemma 3.13** ($\star$). *Let (G, R) be an instance of DSN and H be an inclusion-minimal solution to (G, R) such that every non-terminal vertex in H has at least three neighbors. Let P be a T-avoiding directed path in H between two terminals s and t. Let $p_i, p_j \in Q_P \cup I_P$ with $i < j$ such that there is no $p \in Q_P \cup I_P$ with $p_i \le_P p \le_P p_j$. Let $F = \{p_{i+1}, \dots, p_{j-1}\}$ and let C be the set of vertices of the connected component of* $\mathrm{sym}(H) - \{p_{i+1}, p_{i+2}, p_{j-2}, p_{j-1}\}$ *containing p_{i+3}. If $j - i \ge 5$, then $H[C \cup \{p_{i+1}, p_{i+2}, p_{j-2}, p_{j-1}\}]$ is a ladder and furthermore, $p_{i+1}, p_{i+2}, p_{j-2}$, and p_{j-1} are the only vertices with an H-neighbor outside the ladder.*

▶ **Lemma 3.14** ($\star$). *Let (G, R) be an instance of DSN and H be an inclusion-minimal solution to (G, R) such that H is connected and every non-terminal vertex in H has at least three neighbors. Moreover, assume that for every $s, t \in T$ with $s \ne t$ that $st \in A(R)$ if and only if there is a T-avoiding s-t path in H. Let a, b, c, d be four vertices of H and $F \subseteq V(H)$ such that $a = b$ or $ab \in A(H)$, $c = d$ or $cd \in A(H)$, $F \cap T = \emptyset$, $H[F]$ is a connected component of $H - \{a, b, c, d\}$, and $H[F \cup \{a, b, c, d\}]$ is isomorphic to a ladder $G_{\eta,I}$. There exist a directed graph H' and a set $F' \subseteq V(H')$ such that:*

(1) *the genus of H' is at most the genus of H,*
(2) *$H' - F' = H - F$,*
(3) *$|F'| = O(1)$,*
(4) *$N_{H'}(F') = \{a, b, c, d\}$,*

(5) *H' is an inclusion-minimal solution to R,*
(6) *for every $k \geq 10$, if $\mathrm{sym}(H)$ contains $k \times k$ grid as a minor, then $\mathrm{sym}(H')$ contains $k \times k$ grid as a minor,*
(7) *H' is connected,*
(8) *every non-terminal vertex in H' has at least three neighbors, and*
(9) *for every $s, t \in T$ with $s \neq t$ we have $st \in A(R)$ if and only if there is a T-avoiding s-t path in H'.*

Proof sketch. From Lemma 3.12 it follows that $H[F \cup \{a,b,c,d\}]$ is a union of two directed paths P_1 from a to d and P_2 from c to b. We construct F' such that $H'[F' \cup \{a,b,c,d\}]$ is a ladder $G_{\eta',I'}$, where $I' \subseteq \{1,\eta'\}$ and $1 \in I'$ iff $a = b$ and $\eta' \in I'$ iff $c = d$. Even though it is a bit technical, it is rather straightforward to verify that if we replace F by another ladder, then H' will satisfy (5), (7), (8), and (9). If $\mathrm{sym}(H)$ does not contain any $k \times k$ grid for $k \geq 10$, then we just replace F with any constant size ladder and we are fine. Otherwise, we take a largest grid minor K of $\mathrm{sym}(H)$. Since $\mathrm{sym}(H)[F \cup \{a,b,c,d\}]$ has treewidth 2 and only 4 of its vertices have neighbors in the rest of H, one can show that $\mathrm{sym}(H)[F \cup \{a,b,c,d\}]$ contracts to at most ten vertices in K. Let K_F be the graph induced on these ten vertices. It is easy to see that if we replace $H[F \cup \{a,b,c,d\}]$ with any ladder whose underlying undirected graph has K_F as a minor which furthermore maps its boundaries onto K_F in the same way as $\mathrm{sym}(H)[F \cup \{a,b,c,d\}]$, then the underlying undirected graph of the resulting graph contains K as a minor as well. However, one can express by a constant-sized MSO formula that a boundaried graph is a ladder $G_{\eta',\emptyset}$ and has the boundaried graph K_F as a minor. It follows that this formula has a constant-sized model, whose suitable orientation is the sought replacement. ◀

▶ **Lemma 3.15** (⋆)**.** *Let (G,R) be an instance of DSN and H be an inclusion-minimal solution to (G,R) such that H is connected and every non-terminal vertex in H has at least three neighbors. Moreover, assume that for every $s,t \in T$ with $s \neq t$ that $st \in A(R)$ if and only if there is a T-avoiding s-t path in H. There exists a directed graph H' such that*

- *(H,H') is a $\left(20^{4(2g+3)}\right)$-admissible pair,*
- *$T \subseteq V(H')$,*
- *for all $s,t \in T$, there is a directed s-t path in $H - (T \setminus \{s,t\})$ if and only if there is a directed path from s to t in $H' - (T \setminus \{s,t\})$,*
- *H' is an inclusion-minimal solution to R,*
- *H' is connected,*
- *every non-terminal vertex in H' has at least three neighbors, and*
- *for any arc $st \in A(R)$, there is a T-avoiding directed path P from s to t in H' with length at most $O(|I_P|)$.*

Proof sketch. We obtain H' by recursively applying Lemma 3.14 until there is no ladder with the boundary of size at most 4 that can be shortened by applying Lemma 3.14. By Lemma 3.13 the distance between any two consecutive $p_i, p_j \in Q_P \cup I_P$ is constant. Since the genus of $\mathrm{sym}(H)$ is at most g, it follows from Proposition 2.3 that $\mathrm{sym}(H)$ is $K_{3,2g+3}$-minor-free. Hence, due to Proposition 2.2, the treewidth of $\mathrm{sym}(H)$ is at most $20^{4(2g+3)}\ell$, where ℓ is the size of the largest grid minor of $\mathrm{sym}(H)$ which is the same as of $\mathrm{sym}(H')$ by Lemma 3.14. ◀

Proof of Theorem 3.1. Let H_1 be any connected component of H, $T_1 = V(H_1) \cap T$, and $R_1 = R[T_1]$. By Lemma 3.2, there is R_2 such that for every $s,t \in T_1$ with $s \neq t$ we have $st \in A(R_2)$ if and only if there is a T_1-avoiding s-t path in H_1. By Lemma 3.4, there is a directed graph H_2, such that H_2 is an inclusion-minimal solution to R_2, H_2 is connected, for every $s, t \in T_1$ with $s \neq t$ we have $st \in A(R_2)$ if and only if there is a T_1-avoiding s-t path in H_2, every non-terminal vertex in H_2 has at least three neighbors in H_2 and the genus of H_2 is at most the genus of H_1. By Lemma 3.15, there exists a directed graph H' such that H' is an inclusion-minimal solution to R_2, H' is connected, $\mathrm{tw}(\mathrm{sym}(H_2)) \leq 20^{4(2g+3)}\mathrm{tw}(\mathrm{sym}(H'))$, and for each arc $st \in A(R_2)$, there is a directed path from s to t of length at most $O(|I_P|)$ in H'. Furthermore, all the vertices of H' are on some path of length at most $O(|I_P|)$ between two terminals in H'. By Lemma 3.8, it follows that there is a path of length at most $O(q)$ between each pair of terminals in $\mathrm{sym}(H')$ and hence the diameter of $\mathrm{sym}(H')$ is also at most $O(q)$. Finally, by Proposition 2.4, it follows that $\mathrm{sym}(H')$ has treewidth $O(g'q)$, where g' is the genus of $\mathrm{sym}(H')$. Since the genus of $\mathrm{sym}(H')$ is at most the genus of $\mathrm{sym}(H_2)$, which in turn is at most the genus of $\mathrm{sym}(H_1)$, which in turn is at most the genus of $\mathrm{sym}(H)$, which is at most g, the genus of G, the theorem follows. ◀

4 Improved ETH-based Lower Bound for General Graphs

Our proof is based on a reduction from (a special case of) the following problem:

<table>
<tr><td colspan="2">PARTITIONED SUBGRAPH ISOMORPHISM (PSI)</td></tr>
<tr><td>Input:</td><td>Two undirected graphs G and H with $|V(H)| \leq |V(G)|$ (H is smaller) and a mapping $\psi\colon V(G) \to V(H)$.</td></tr>
<tr><td>Question:</td><td>Is H isomorphic to a subgraph of G? I.e., is there an injective mapping $\phi\colon V(H) \to V(G)$ such that $\{\phi(u), \phi(v)\} \in E(G)$ for each $\{u, v\} \in E(H)$ and $\psi \circ \phi$ is the identity?</td></tr>
</table>

▶ **Theorem 4.1** (Marx [32, Corollary 6.1]). *If there exist a recursively enumerable class $\mathcal{H}$ of graphs with unbounded treewidth, an algorithm $\mathbb{A}$, and an arbitrary function f such that $\mathbb{A}$ correctly decides every instance of* PARTITIONED SUBGRAPH ISOMORPHISM *with the smaller graph H in $\mathcal{H}$ in time $f(H)n^{o(\mathrm{tw}(H)/\log \mathrm{tw}(H))}$, then ETH fails.*

It is known that there are infinitely many 3-regular graphs such that each such graph H has treewidth $\Theta(|V(H)|)$ (see [23, Proposition 1, Theorem 5]). Using the class of 3-regular graphs as $\mathcal{H}$ in the above theorem, we arrive at the following corollary.

▶ **Corollary 4.2.** *If there is an algorithm $\mathbb{A}$ and an arbitrary function f such that $\mathbb{A}$ correctly decides every instance of* PARTITIONED SUBGRAPH ISOMORPHISM *with the smaller graph H being 3-regular in time $f(|V(H)|)n^{o(|V(H)|/\log |V(H)|)}$, then ETH fails.*

Our plan is to use this corollary. To this end, we transform the (special) instances of PSI to instances of DSN.

▶ **Construction 1.** Let (G, H, ψ) be an instance of PSI with H 3-regular and denote $k = |V(H)|$. Note that then $|E(H)| = O(k)$. We let $r = \lceil \sqrt{k} \rceil$. We first compute labelings $\alpha\colon V(H) \to X$, $\beta\colon V(H) \to Y$, and $\gamma\colon E(H) \to Z$, where $X = \{x_1, \ldots, x_{\max}\}$, $Y = \{y_1, \ldots, y_{\max}\}$, and $Z = \{z_1, \ldots, z_{\max}\}$ are three new sets. We want the sets X, Y, Z to be of size $O(r)$ while fulfilling the following constraints:

(i) $\forall u, v \in V(H) : (\alpha(u) \neq \alpha(v)) \vee (\beta(u) \neq \beta(v))$,
(ii) $\forall \{u, v\} \in E(H) : (\alpha(u) \neq \alpha(v)) \wedge (\beta(u) \neq \beta(v))$,
(iii) $\forall e, f \in E(H), \forall u, v \in V(H) : ((u \in e) \wedge (v \in f) \wedge (\alpha(u) = \alpha(v))) \implies (\gamma(e) \neq \gamma(f))$.

In other words, the pair $(\alpha(u), \beta(u))$ uniquely identifies vertex u, adjacent vertices share no labels and both pairs $(\alpha(u), \gamma(\{u, v\}))$ and $(\alpha(v), \gamma(\{u, v\}))$ uniquely identify edge $\{u, v\}$.

To obtain such labeling, first color the vertices of H greedily with colors $1,\ldots,4$, denote μ the coloring and $A_1,\ldots,A_4$ the set of vertices of color $1,\ldots,4$, respectively. For every $i\in[4]$, we split the set A_i into sets $A_{i,1},\ldots,A_{i,a_i}$ such that for every $j\in[a_i-1]$ the set $A_{i,j}$ is of the size r and the set A_{i,a_i} is of the size at most r. Since $r=\left\lceil\sqrt{k}\right\rceil$ we know that there will be at most r sets of the size r and, thus, at most $r+4$ sets in total. We assign to each nonempty set $A_{i,j}$ a unique label x_ℓ and let $\alpha(u)=x_\ell$ for every $u\in A_{i,j}$. Note that $|X|\le r+4$.

Next we construct a graph H' from H by turning each $A_{i,j}$ into a clique. Since the degree of each vertex in H is 3 and the size of each $A_{i,j}$ is at most r, the degree of each vertex in H' is at most $r+2$. Hence we can color the vertices of H' greedily with colors $y_1,\ldots,y_{r+3}$ and we let β be the coloring.

Finally, we construct a multigraph H'' from H' by contracting each clique $A_{i,j}$ to a single vertex. We keep multiple edges between two vertices if they are a result of the contraction, but we remove all loops. Note that the edges preserved are exactly the edges of H. Since the size of each $A_{i,j}$ is at most r and H is 3-regular, the maximum degree (counting the multiplicities of the edges) is at most $3r$. Therefore, the maximum degree in the line graph $L(H'')$ of H'' is at most $6r-2$. Thus, we can color the edges of H'' greedily with colors $z_1,\ldots,z_{6r-1}$ and let γ be the coloring.

Let us check that the labelings fulfill the constraints. First, if $\alpha(u)=\alpha(v)$, then $\{u,v\}\in E(H')$ and, thus, $\beta(u)\neq\beta(v)$. If $\{u,v\}\in E(H)$, then $\{u,v\}\subseteq A_{i,j}$ would imply that u and v are colored by the same color by μ – a contradiction. Hence, $\alpha(u)\neq\alpha(v)$ and, since $E(H)\subseteq E(H')$, we also have $\beta(u)\neq\beta(v)$. Finally, if $e=\{u,v'\}$, $f=\{u',v\}$, and $\alpha(u)=\alpha(v)$, then the edges e and f share a vertex in H'' and, thus, $\gamma(e)\neq\gamma(f)$.

Note also that the labelings can be obtained in $O(|V(H)|^2)$ time.

Having the labelings at hand, we construct the instance (G',R) of DSN as follows (refer to Figure 4 for an overview of the construction). We let $V(G')=V\cup W\cup X\cup Y\cup Z$, where $V=V(G)$, $W=\{w_{uv}\mid\{u,v\}\in E(G)\}$, and X,Y,Z are the images of α,β,γ as defined previously. We let $T=V(R)=X\cup Y\cup Z$. Note that $q=O(r)=O(\sqrt{k})$. We let $A(G')=A_V\cup A_W$, where $A_V=\big\{\big(\alpha(\psi(u)),u\big),\big(u,\beta(\psi(u))\big)\mid u\in V\big\}$ and $A_W=\big\{(u,w_{uv}),(v,w_{uv}),\big(w_{uv},\gamma(\{\psi(u),\psi(v)\})\big)\mid\{u,v\}\in E(G)\big\}$. We assign unit weights to all arcs of G'. Finally let $A(R)=A_Y\cup A_Z$, where $A_Y=\{(\alpha(u),\beta(u))\mid u\in V(H)\}$ and $A_Z=\{(\alpha(u),\gamma(\{u,v\})),(\alpha(v),\gamma(\{u,v\}))\mid\{u,v\}\in E(H)\}$.

Let us stop here to discuss the size of $A(R)$. By Condition (i) on the labelings we have $|A_Y|=|V(H)|$. By Condition (ii) we have $(\alpha(u),\gamma(\{u,v\}))\neq(\alpha(v),\gamma(\{u,v\}))$ for any $\{u,v\}\in E(H)$. Hence, by Condition (iii) the size of A_Z is exactly $2|E(H)|$.

Next, we show that the construction transforms yes-instances of PSI to instances of DSN with bounded value of the optimum.

▶ **Lemma 4.3.** *If there is an injective mapping ϕ forming a solution to the instance (G,H,ψ) of PSI, then there is a subgraph P of G' forming a solution to the instance (G',R) of DSN with cost $|A(P)|\le 2|V(H)|+3|E(H)|$.*

Proof. Let ϕ be a solution to the instance (G,H,ψ). Since ϕ is a solution, we know that $\{\phi(u),\phi(v)\}\in E(G)$ whenever $\{u,v\}\in E(H)$. Consider the subgraph $P=G'[V_\phi]$ of G' induced by $V_\phi=X\cup Y\cup Z\cup V'\cup W'$, where $V'=\{\phi(v)\mid v\in V(H)\}$ and $W'=\big\{w_{\phi(u)\phi(v)}\mid\{u,v\}\in E(H)\big\}$. Obviously, $|V'|=|V(H)|$ and $|W'|=|E(H)|$.

Since each arc in A_W is incident to some vertex in W and each vertex in W is incident to exactly 3 such arcs, P contains at most $3|E(H)|$ arcs from A_W. Similarly, since each arc in A_V is incident to some vertex in V and each vertex in V is incident to exactly 2 such arcs, P contains at most $2|V(H)|$ arcs from A_V. Thus, P contains at most $2|V(H)|+3|E(H)|$ arcs in total.

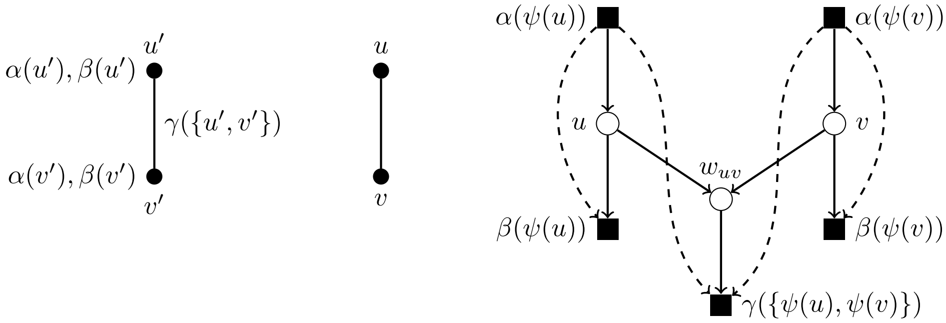

Figure 4 An illustration of Construction 1. Left is a pattern graph H, middle a host graph G, and right the produced graphs G' and R combined. We assume $\psi(u) = u'$ and $\psi(v) = v'$ here. On the right the terminals are depicted by full squares and non-terminals by empty circles. Arcs in G' are drawn solid, while the arcs of R are dashed.

We want to show for each $(s,t) \in A(R)$ that there is a directed path from s to t in P. Indeed, if $(x,y) \in A_Y$, then $x = \alpha(u)$ and $y = \beta(u)$ for some $u \in V(H)$ and $(\alpha(u), \phi(u), \beta(u)) = (\alpha(\psi(\phi(u))), \phi(u), \beta(\psi(\phi(u))))$ is a path of length 2 from x to y in P. If $(x,z) \in A_Z$, then $x = \alpha(u)$ and $z = \gamma(\{u,v\})$ for some $\{u,v\} \in E(H)$ and $(\alpha(u), \phi(u), w_{\phi(u)\phi(v)}, \gamma(\{u,v\}))$ is a path of length 3 from x to z in P. This finishes the proof. ◀

Next we show that the value of the optimum of the instances of DSN produced by the construction can be appropriately bounded only if we started with a yes-instance of PSI.

▶ **Lemma 4.4** (⋆). *If there is a subgraph P of G' forming a solution to the instance (G', R) of DSN with cost $|A(P)| \leq 2|V(H)| + 3|E(H)|$, then there is an injective mapping ϕ forming a solution to the instance (G, H, ψ) of PSI.*

Proof of Theorem 1.2. Let $\mathbb{A}$ be an algorithm that correctly solves DSN (on general graphs) in time $f(q)n^{o(q^2/\log q)}$ for some function f. Let us construct an algorithm $\mathbb{B}$ for PSI with the smaller graph H being 3-regular as follows: Let (G, H, ψ) be an instance of PSI with H 3-regular. We use Construction 1 to build the instance (G', R) of DSN. Then run $\mathbb{A}$ on (G', R) and return yes if and only if the cost of the obtained solution P is $|A(P)| \leq 2|V(H)| + 3|E(H)|$. The answer of $\mathbb{B}$ is correct by Lemmata 4.3 and 4.4.

Let us analyze the running time of $\mathbb{B}$. Let us denote $k = |V(H)|$ and $n = |V(G)|$. We may assume that $k \leq n$, as otherwise we can immediately answer no. The labelings can be obtained in $O(k^2)$ time. Graph G has at most $O(n^2)$ edges and the graphs G' and R can be constructed in linear time in the number of vertices and edges of the graphs G and H, respectively. That is, Construction 1 can be performed in $O(n^2)$ time and, in particular, G' has $O(n^2)$ vertices. However, by the construction, the number q of vertices of graph R is $O(\sqrt{k})$. Now, $\mathbb{A}$ runs on (G', R) in time $f(q)|V(G')|^{o(q^2/\log q)} = f'(\sqrt{k})n^{o((\sqrt{k})^2/\log \sqrt{k})} = f''(k)n^{o(k/\log k)}$ for some functions f, f', and f''. But then the whole $\mathbb{B}$ runs in $f''(k)n^{o(k/\log k)}$ time and ETH fails by Corollary 4.2. ◀

5 Conclusions

Our results show that we can solve DSN in time $n^{O(q)}$ when the input directed graph is embeddable on a fixed surface However, for general graphs it is unlikely to obtain even an algorithm running in time $n^{o(q^2/\log(q))}$. It would be interesting to see what happens for the graph classes that are somewhere in between. For example, it is not difficult to show that the graph H' that we obtain in Section 3 has at most $O(q^3)$ vertices and, hence, the largest grid minor of H' is of size $O(q^{3/2}) \times O(q^{3/2})$. Therefore, with a careful modification of our approach, one can show that there is an $n^{O(q^{3/2})}$ time algorithm for DSN when the input graph excludes a fixed minor. However, it remains open whether the running time $n^{O(q^{3/2})}$ is asymptotically optimal or whether it is possible to design an $n^{O(q)}$ time algorithm for DSN in this case.

References

1 Marshall W. Bern and Paul E. Plassmann. The Steiner Problem with Edge Lengths 1 and 2. *Inf. Process. Lett.*, 32(4):171–176, 1989. doi:10.1016/0020-0190(89)90039-2.

2 Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Fourier meets Möbius: fast subset convolution. In *Proceedings of the 39th ACM Symposium on Theory of Computing, STOC 2007*, pages 67–74. ACM, 2007.

3 Jarosław Byrka, Fabrizio Grandoni, Thomas Rothvoß, and Laura Sanità. Steiner Tree Approximation via Iterative Randomized Rounding. *J. ACM*, 60(1):6:1–6:33, February 2013. doi:10.1145/2432622.2432628.

4 Moses Charikar, Chandra Chekuri, To-Yat Cheung, Zuo Dai, Ashish Goel, Sudipto Guha, and Ming Li. Approximation Algorithms for Directed Steiner Problems. *Journal of Algorithms*, 33(1):73–91, 1999. doi:10.1006/jagm.1999.1042.

5 Chandra Chekuri, Guy Even, Anupam Gupta, and Danny Segev. Set connectivity problems in undirected graphs and the directed Steiner network problem. *ACM Trans. Algorithms*, 7(2):18:1–18:17, 2011. doi:10.1145/1921659.1921664.

6 Rajesh Chitnis, Andreas Emil Feldmann, and Pasin Manurangsi. Parameterized Approximation Algorithms for Bidirected Steiner Network Problems. In Yossi Azar, Hannah Bast, and Grzegorz Herman, editors, *26th Annual European Symposium on Algorithms (ESA 2018)*, volume 112 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20:1–20:16, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.ESA.2018.20.

7 Rajesh Chitnis, Mohammadtaghi Hajiaghayi, and Daniel Marx. Tight Bounds for Planar Strongly Connected Steiner Subgraph with Fixed Number of Terminals (and Extensions). In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1782–1801. SIAM, 2014. doi:10.1137/1.9781611973402.129.

8 Dietmar Cieslik. *Steiner minimal trees*, volume 23 of *Nonconvex Optimization and Its Applications*. Springer Science & Business Media, 1998.

9 Marek Cygan, Holger Dell, Daniel Lokshtanov, Dániel Marx, Jesper Nederlof, Yoshio Okamoto, Ramamohan Paturi, Saket Saurabh, and Magnus Wahlström. On Problems as Hard as CNF-SAT. *ACM Trans. Algorithms*, 12(3):41, 2016. doi:10.1145/2925416.

10 Erik D. Demaine, MohammadTaghi Hajiaghayi, and Ken-ichi Kawarabayashi. Algorithmic Graph Minor Theory: Improved Grid Minor Bounds and Wagner's Contraction. *Algorithmica*, 54(2):142–180, 2009. doi:10.1007/s00453-007-9138-y.

11 Reinhard Diestel. *Graph Theory, 5th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2017.

12 Irit Dinur and Pasin Manurangsi. ETH-Hardness of Approximating 2-CSPs and Directed Steiner Network. In Anna R. Karlin, editor, *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*, volume 94 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 36:1–36:20, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.ITCS.2018.36.

13 Yevgeniy Dodis and Sanjeev Khanna. Design Networks with Bounded Pairwise Distance. In *Proc. 31th STOC*, pages 750–759. ACM, 1999.

14 Stuart E. Dreyfus and Robert A. Wagner. The Steiner problem in graphs. *Networks*, 1:195–207, 1972.

15 David Eppstein. Diameter and Treewidth in Minor-Closed Graph Families. *Algorithmica*, 27(3):275–291, 2000. doi:10.1007/s004530010020.

16 Ranel E. Erickson, Clyde L. Monma, and Arthur F. Veinott, Jr. Send-and-Split Method for Minimum-Concave-Cost Network Flows. *Mathematics of Operations Research*, 12(4):634–664, 1987. doi:10.1287/moor.12.4.634.

17 Jon Feldman and Matthias Ruhl. The Directed Steiner Network Problem Is Tractable for a Constant Number of Terminals. *SIAM Journal on Computing*, 36(2):543–561, 2006.

18 Andreas Emil Feldmann and Dániel Marx. The Complexity Landscape of Fixed-Parameter Directed Steiner Network Problems. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*, volume 55 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 27:1–27:14, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.ICALP.2016.27.

19 Andreas Emil Feldmann and Dániel Marx. The Complexity Landscape of Fixed-Parameter Directed Steiner Network Problems. *CoRR*, abs/1707.06808, 2017. arXiv:1707.06808.

20 Fedor V. Fomin, Petteri Kaski, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Parameterized Single-Exponential Time Polynomial Space Algorithm for Steiner Tree. In *Automata, Languages, and Programming - 42nd International Colloquium, ICALP 2015, Proceedings, Part I*, volume 9134 of *LNCS*, pages 494–505. Springer, 2015. doi:10.1007/978-3-662-47672-7_40.

21 Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, and Saket Saurabh. Planar F-Deletion: Approximation, Kernelization and Optimal FPT Algorithms. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 470–479. IEEE Computer Society, 2012. doi:10.1109/FOCS.2012.62.

22 Bernhard Fuchs, Walter Kern, Daniel Mölle, Stefan Richter, Peter Rossmanith, and Xinhui Wang. Dynamic Programming for Minimum Steiner Trees. *Theory of Computing Systems*, 41(3):493–500, 2007. doi:10.1007/s00224-007-1324-4.

23 Martin Grohe and Dániel Marx. On tree width, bramble size, and expansion. *J. Combin. Theory Ser. B*, 99(1):218–228, 2009. doi:10.1016/j.jctb.2008.06.004.

24 Jonathan L. Gross and Thomas W. Tucker. *Topological Graph Theory*. Wiley-Interscience, New York, NY, USA, 1987.

25 Jiong Guo, Rolf Niedermeier, and Ondřej Suchý. Parameterized Complexity of Arc-Weighted Directed Steiner Problems. *SIAM Journal on Discrete Mathematics*, 25(2):583–599, 2011.

26 Eran Halperin and Robert Krauthgamer. Polylogarithmic Inapproximability. In *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 585–594, New York, NY, USA, 2003. ACM. doi:10.1145/780542.780628.

27 Frank K. Hwang, Dana S. Richards, and Pawel Winter. *The Steiner Tree Problem*, volume 53 of *Annals of Discrete Mathematics*. Elsevier, 1992. doi:10.1016/S0167-5060(08)70188-2.

28 Russell Impagliazzo and Ramamohan Paturi. On the Complexity of k-SAT. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.

29 Richard M. Karp. Reducibility among Combinatorial Problems. In Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger, editors, *Proceedings of a symposium on the Complexity of Computer Computations, 1972*, pages 85–103, Boston, MA, 1972. Springer US. doi:10.1007/978-1-4684-2001-2_9.

30 Anatolii Y. Levin. Algorithm for the shortest connection of a group of graph vertices. *Sov. Math. Dokl.*, 12:1477–1481, 1971.

31 Leonid Libkin. *Elements of Finite Model Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2004. doi:10.1007/978-3-662-07003-1.

32 Dániel Marx. Can You Beat Treewidth? *Theory of Computing*, 6(1):85–112, 2010. doi:10.4086/toc.2010.v006a005.

33 Bojan Mohar and Carsten Thomassen. *Graphs on Surfaces*. Johns Hopkins series in the mathematical sciences. Johns Hopkins University Press, 2001.

34 Jesper Nederlof. Fast Polynomial-Space Algorithms Using Inclusion-Exclusion. *Algorithmica*, 65(4):868–884, 2013. doi:10.1007/s00453-012-9630-x.

35 Hans Jürgen Prömel and Angelika Steger. *The Steiner Tree Problem; a Tour through Graphs, Algorithms, and Complexity*. Vieweg, 2002.

36 Ondřej Suchý. On Directed Steiner Trees with Multiple Roots. In Pinar Heggernes, editor, *Graph-Theoretic Concepts in Computer Science - 42nd International Workshop, WG 2016, Istanbul, Turkey, June 22-24, 2016, Revised Selected Papers*, volume 9941 of *Lecture Notes in Computer Science*, pages 257–268, 2016. doi:10.1007/978-3-662-53536-3_22.