

Secluded Connectivity Problems

Shiri Chechik ^{*} M. P. Johnson [†] Merav Parter ^{‡§} David Peleg ^{‡ ¶}

December 13, 2012

Abstract

Consider a setting where possibly sensitive information sent over a path in a network is visible to every neighbor of the path, i.e., every neighbor of some node on the path, thus including the nodes *on* the path itself. The *exposure* of a path P can be measured as the number of nodes adjacent to it, denoted by $N[P]$. A path is said to be *secluded* if its exposure is small. A similar measure can be applied to other connected subgraphs, such as Steiner trees connecting a given set of terminals. Such subgraphs may be relevant due to considerations of privacy, security or revenue maximization. This paper considers problems related to minimum exposure connectivity structures such as paths and Steiner trees. It is shown that on unweighted undirected n -node graphs, the problem of finding the minimum exposure path connecting a given pair of vertices is strongly inapproximable, i.e., hard to approximate within a factor of $O(2^{\log^{1-\epsilon} n})$ for any $\epsilon > 0$ (under an appropriate complexity assumption), but is approximable with ratio $\sqrt{\Delta} + 3$, where Δ is the maximum degree in the graph. One of our main results concerns the class of bounded-degree graphs, which is shown to exhibit the following interesting dichotomy. On the one hand, the minimum exposure path problem is NP-hard on *node-weighted* or *directed* bounded-degree graphs (even when the maximum degree is 4). On the other hand, we present

^{*}Microsoft Research Silicon Valley Center, USA. Email: schechik@microsoft.com

[†]Department of Electrical Engineering, UCLA, Los Angeles, USA. Email: mpjohnson@gmail.com

[‡]The Weizmann Institute of Science, Rehovot, Israel. Email: {merav.parter,david.peleg}@weizmann.ac.il.

[§]Recipient of the Google Europe Fellowship in distributed computing; research supported in part by this Google Fellowship.

[¶]Supported in part by the Israel Science Foundation (grant 894/09), the United States-Israel Binational Science Foundation (grant 2008348), the Israel Ministry of Science and Technology (infrastructures grant), and the Citi Foundation.

a polynomial algorithm (based on a nontrivial dynamic program) for the problem on unweighted undirected bounded-degree graphs. Likewise, the problem is shown to be polynomial also for the class of (weighted or unweighted) bounded-treewidth graphs. Turning to the more general problem of finding a minimum exposure Steiner tree connecting a given set of k terminals, the picture becomes more involved. In undirected unweighted graphs with unbounded degree, we present an approximation algorithm with ratio $\min\{\Delta, n/k, \sqrt{2n}, O(\log k \cdot (k + \sqrt{\Delta}))\}$. On unweighted undirected bounded-degree graphs, the problem is still polynomial when the number of terminals is fixed, but if the number of terminals is arbitrary, then the problem becomes NP-hard again.

1 Introduction

The problem. Consider a setting where possibly sensitive information sent over a path in a network is visible to every neighbor of the path, i.e., every neighbor of some node on the path, thus including the nodes *on* the path itself. The *exposure* of a path P can be measured as the size (possibly node-weighted) of its *neighborhood* in this sense, denoted by $N[P]$. A path is said to be *secluded* if its exposure is small. A similar measure can be applied to other connected subgraphs, such as Steiner trees connecting a given set of terminals. Our interest is in finding connectivity structures with exposure as low as possible. This may be motivated by the fact that in real-life applications, a connectivity structure operates normally as part of the entire network G (and is not “extracted” from it), and so controlling the effect of its operation on the other nodes in the network may be of interest, in situations in which any “activation” of a node (by taking it as part of the structure) leads to an activation of its neighbors as well. In such settings, to minimize the set of total active nodes, we aim toward finding secluded or sufficiently private connectivity structures. Such subgraphs may be important in contexts where privacy is an important concern, or in settings where security measures must be installed on any node from which the information is visible, making it desirable to minimize their number. Another context where minimizing exposure may be desirable is when the information transferred among the participants has commercial value and overexposure to “free viewers” implies revenue loss.

This paper considers the problem of minimizing the exposure of subgraphs that satisfy some desired connectivity requirements. Two fundamental connectivity problems are considered, namely, single-path connectivity and Steiner trees, formulated as the **Secluded**

Path and **Secluded Steiner Tree** problems, respectively, as follows. Given a graph $G = (V, E)$ and an s, t pair (respectively, a terminal set \mathcal{S}), it is required to find an $s - t$ path (respectively, a Steiner tree) of minimum exposure.

Related Work. The problems considered in this paper are variations of the classical shortest path and Steiner tree problems. In the standard versions of these problems, a cost measure is associated with edges or vertices, e.g., representing length or weight and the task is to identify a minimum cost subgraph satisfying the relevant connectivity requirement. Essentially, the cost of the solution subgraph is a *linear* sum of the solution’s *constituent parts*, i.e., the sum of the weights of the edges or vertices chosen.

In contrast, in the setting of *labeled connectivity* problems, edges (and occasionally vertices) are associated with *labels* (or *colors*) and the objective is to identify a subgraph $G' \subseteq G$ that satisfies the connectivity requirements while minimizing the number of used labels. In other words, costs are now assigned to labels rather than to single edges. Such labeling schemes incorporate grouping constraints, based on partitioning the set of available edges into classes, each of which can be purchased in its entirety or not at all. These grouping constraints are motivated by applications from telecommunication networks, electrical networks, and multi-modal transportation networks. Labeled connectivity problems have been studied extensively from complexity-theoretic and algorithmic points of view [7, 23, 11, 9]. The optimization problems in this category include, among others, the **Minimum Labeled Path** problem [11, 23], the **Minimum Labeled Spanning Tree** problem [15, 11], the **Minimum Labeled Cut** problem [24], and the **Labelled Perfect Matching** problem [20].

In both the traditional setting and the labeled connectivity setting, only edges or nodes that are explicitly part of the selected output structure are “paid for” in solution cost. That is, the cost of a candidate structure is a pure function of its components, ignoring the possible effects of “passive” participants, such as nodes that are “very close” to the structure in the input graph G . In contrast, in the setting considered in this paper, the cost of a connectivity structure G' is a function not only of its components but also of their immediate surroundings, namely, the manner in which G' is embedded in G plays a role as well. (Alternatively, we can say that the cost is a not necessarily a linear function of its components.)

To the best of our knowledge, secluded connectivity problems have not been considered before in the literature. The **Secluded Path** and **Secluded Steiner Tree** problems are related to several existing combinatorial optimization problems. These include the

Red-Blue Set Cover problem [4, 21], the Minimum Labeled Path problem [11, 23] and the Steiner Tree [13] and Node Weighted Steiner Tree problems [14]. A prototypical example is the Red-Blue Set Cover problem, in which we are given a set R of red elements, a set B of blue elements and a family $S \subseteq 2^{R \cup B}$ of subsets of blue and red elements, and the objective is to find a subfamily $C \subseteq S$ covering all blue elements that minimizes the number of red elements covered. This problem is known to be strongly inapproximable.

Finally, turning to geometric settings, similarly motivated problems have been studied in the networking and sensor networks communities, where sensors are often modeled as unit disks. For example, the Maximal Breach Path problem [19] is defined in the context of traversing a region of the plane that contains sensor nodes at predetermined points, and its objective is to maximize the minimum distance between the points on the path and the the sensor nodes. (The solution uses edges of the sensor nodes' Voronoi diagram.) A dual problem studied extensively is *barrier coverage*, i.e., the (deterministic or stochastic) placement of sensors in order to make it difficult for an adversary to cross the region unseen (see [16] and the references therein). [5] studies k -barrier coverage, in which the task is to position the sensors so that any path across the region will intersect with at least k sensor disks. Similarly motivated problems have been studied in the context of path planning in AI. “Stealth” path planning problems, in which the task is to find a minimum “visibility” path from source to destination, have been considered in [12, 17, 18]. Although the motivation is similar, such problems are technically quite different from the graph-based problems studied here; those problems are typically posed in the geometric plane, amid obstacles that cause occlusion, and visibility is defined in terms of line-of-sight.

Contributions. In this paper, we introduce the concept of *secluded connectivity* and study some of its complexity and algorithmic aspects. We first show that the **Secluded Path** (and hence also **Secluded Steiner Tree**) problem is strongly inapproximable on unweighted undirected graphs with unbounded degree (more specifically, is hard to approximate with ratio $O(2^{\log^{1-\epsilon} n})$, where n is the number of nodes in the graph G , assuming $\mathcal{NP} \not\subseteq \mathcal{DTIME}(n^{\text{poly} \log n})$). Conversely, we devise a $\sqrt{\Delta} + 3$ approximation algorithm for the **Secluded Path** problem and a $\min\{\Delta, n/k, \sqrt{2n}, O(\log k \cdot (k + \sqrt{\Delta}))\}$ approximation algorithm for the **Secluded Steiner Tree** problem, where Δ is the maximum degree in the graph and k is the number of terminals.

One of our key results concerns bounded-degree graphs and reveals an interesting dichotomy. On the one hand, we show that **Secluded Path** is NP-hard on the class of *node-weighted* or *directed* bounded-degree graphs, even if the maximum degree is 4. In

contrast, we show that on the class of unweighted undirected bounded-degree graphs, the **Secluded Path** problem admits an *exact* polynomial-time algorithm, which is based on a complex dynamic programming and requires some nontrivial analysis. Likewise, the **Secluded Steiner Tree** problem with fixed size terminal set is in P as well.

Finally, we consider some specific graph classes. We show that the **Secluded Path** and **Secluded Steiner Tree** problems are polynomial for bounded-treewidth graphs. We also show that the **Secluded Path** (resp., **Secluded Steiner Tree**) problem can be approximated with ratio $O(1)$ (resp., $\Theta(\log k)$) in polynomial time for hereditary graph classes of bounded density. As an example, the **Secluded Path** problem has a 6 approximation on planar graphs. (A more careful direct analysis of the planar case yields ratio 3.)

Preliminaries. Consider a node-weighted graph $G(V, E, W)$, for some weight function $W : V \rightarrow \mathbb{R}_{\geq 0}$, with n nodes and maximum degree Δ . For a node $u \in V$, let $N(u) = \{v \in V \mid (u, v) \in E\}$ be the set of u 's neighbors and let $N[u] = N(u) \cup \{u\}$ be u 's *closed neighborhood*, i.e., including u itself. A *path* is a sequence $P = [u_1, \dots, u_\ell]$, oriented from left to right, also termed a $u_1 - u_\ell$ path. Let $P[i] = u_i$ for $i \in \{1, \dots, \ell\}$. Let $First(P) = u_1$ and $Last(P) = u_\ell$. For a path P and nodes x, y on it, let $P[x, y]$ be the subpath of P from x to y . For a connected subgraph $G' \subseteq G$ and for $u_i, u_j \in V(G')$, let $dist_{G'}(u_i, u_j)$ be the distance between u_i and u_j in G' . Let $N(G') = \bigcup_{u \in G'} N(u) \setminus G'$ be the nodes that are strictly neighbors of G' nodes and $N[G'] = \bigcup_{u \in G'} N[u]$ be the set of nodes in the 1-neighborhood of G' . Define the cost of G' as

$$\text{Cost}(G') = \sum_{u \in N[G']} W(u) . \quad (1)$$

Note that if G is unweighted, then the cost of a subgraph G' is simply the cardinality of the set of G' nodes and their neighbors, $\text{Cost}(G') = |N[G']|$.

We sometimes consider the neighbors of node $u \in V(G)$ in different subgraphs. To avoid confusion, we denote $N_{G'}(u)$ the neighbors of u restricted to graph G' .

For a subgraph $G' \subseteq G$, let $\text{DegCost}(G')$ denote the sum of the degrees of the nodes of G' . If G' is a path, then this key parameter is closely related to our problem. It is not hard to see that for any given path P , $\text{Cost}(P) \leq \text{DegCost}(P)$. The problem of finding an $s - t$ path P with minimum $\text{DegCost}(P)$ is polynomial, making it a convenient starting-point for various heuristics for the problem.

In this paper we consider two main connectivity problems. In the **Secluded Path** problem we are given an unweighted graph $G(V, E)$, a source node s and target node t , and the objective is to find an $s - t$ path P with minimum neighborhood size. A

generalization of this problem is the **Secluded Steiner Tree** problem, in which instead of two terminals s and t we are given a set of k terminal nodes \mathcal{S} and it is required to find a tree T in G covering \mathcal{S} , of minimum neighborhood size. If the given graph G is weighted, then the *weighted Secluded Path* and **Secluded Steiner Tree** problems require minimizing the neighborhood cost as given in Eq. (1). We now define these tasks formally. For an $s - t$ pair, let $\mathcal{P}_{s,t} = \{P \mid P \text{ is a } s\text{-}t \text{ path}\}$ be the set of all $s - t$ paths and let $q_{s,t}^* = \min\{\text{Cost}(P) \mid P \in \mathcal{P}_{s,t}\}$ be the minimum cost among these paths. Then the objective of the **Secluded Path** problem is to find a $P^* \in \mathcal{P}_{s,t}$ that attains this minimum, i.e., such that $\text{Cost}(P^*) = q_{s,t}^*$. For the **Secluded Steiner Tree** problem, let $\mathcal{T}(\mathcal{S}) = \{T \subseteq G \mid \mathcal{S} \subseteq V(T), T \text{ is a tree}\}$ be the set of all trees in G covering \mathcal{S} , and let $q^*(\mathcal{S})$ be the minimum cost among these trees, i.e.,

$$q^*(\mathcal{S}) = \min\{\text{Cost}(T) \mid T \in \mathcal{T}(\mathcal{S})\} . \quad (2)$$

Then the solution for the problem is a tree $T^* \in \mathcal{T}(\mathcal{S})$ such that $\text{Cost}(T^*) = q^*(\mathcal{S})$.

Notation For a graph $G = (V, E)$ and a set \mathcal{S} of k terminals, let T be some Steiner tree (spanning all terminals). Consider a decomposition of the tree into $O(k)$ subpaths $\Sigma(T)$ defined as follows. Let $B^*(T) = \mathcal{S} \cup \{u \in V(G) \mid |N_T(u)| \geq 3\}$ be the set of terminals and nodes with more than three neighbors in T (thus branching points in T). For $a, b \in B^*(T)$, let $\alpha = [a, b]$ be some maximal path in T such that $V(\alpha) \cap B^*(T) = \{a, b\}$, i.e., α is a maximal subpath in T whose internal section is *free* of $B^*(T)$ nodes. Since any internal node in α is of degree at most 2, α is indeed a path in T . Let $\Sigma(T) = \{\alpha = [a, b] \mid a, b \in B^*(T)\}$ be the collection of these maximal subpaths. Note that $|\Sigma(T)| \leq 2k$.

2 Unweighted Undirected Graphs with Unbounded Degree

Hardness of approximation.

Theorem 2.1 *On unweighted undirected graphs with unbounded degree, the Secluded Path problem (and hence also the Secluded Steiner Tree problem) is strongly inapproximable. Specifically, unless $\mathcal{NP} \subseteq \text{DTIME}(n^{\text{poly} \log(n)})$, the Secluded Path problem cannot be approximated to within a factor $O(2^{\log^{1-\epsilon} n})$ for any $\epsilon > 0$.*

We show this for the simple case where $\mathcal{S} = \{s, t\}$. The theorem is shown by a gap-preserving reduction from the **Red-Blue Set Cover** (RBSC) problem [21]. An instance

of **Red-Blue Set Cover** consists of a set $B = \{b_1, \dots, b_{|B|}\}$ of blue elements, a set $R = \{r_1, \dots, r_{|R|}\}$ of red elements, and a collection $S = \{S_1, \dots, S_m\}$ of subsets of $B \cup R$. The task is to choose a family of sets covering all blue elements but a minimum number of red elements. In [4] it is shown that the **Red-Blue Set Cover** problem cannot be approximated to within an $O(2^{\log^{1-\epsilon} n})$ ratio unless $\mathcal{NP} \subseteq \text{DTIME}(n^{\text{poly} \log(n)})$, where $n = \max\{|R|, |B|, |S|\}$. Given a **Red-Blue Set Cover** instance, we construct a **Secluded Path** instance (G, s, t) as follows. For each b_i , let $\mathcal{S}_i = \{S_j \in S \mid b_i \in S_j\}$ be the sets containing b_i . For every i, j , add to G a node $v_{i,j}$ corresponding to $S_j \in \mathcal{S}_i$. Define $\mathcal{V}_i = \bigcup_{S_j \in \mathcal{S}_i} \{v_{i,j}\}$ and $\mathcal{V} = \bigcup_i \mathcal{V}_i$. Add edges so that the sets \mathcal{V}_i and \mathcal{V}_{i+1} form a complete bipartite graph for every $1 \leq i < |B|$. In addition, we connect a node s to all nodes in \mathcal{V}_1 , and connect a node t to all nodes in $\mathcal{V}_{|B|}$. So far, our construction contains $X = \sum_i |\mathcal{V}_i| + 2 \leq n^2 + 2$ nodes (see Fig. 1).

We now turn to describing the representation of the red elements R . For every $r_\ell \in R$, let \mathcal{C}_ℓ be a “supernode” consisting of n^3 nodes (with no edges among them), and let $\mathcal{C} = \bigcup_{\ell=1}^{|R|} \mathcal{C}_\ell$ be the set of all supernodes. All n^3 nodes of \mathcal{C}_ℓ are connected to node $v_{i,j}$ if $r_\ell \in S_j$, for every $\ell \in \{1, \dots, |R|\}$, $i \in \{1, \dots, |B|\}$, and $j \in \{1, \dots, |S|\}$. Finally, let \mathcal{H} be a “hypernode” consisting of n^5 nodes (with no edges among them), all of which are connected to all the constituent nodes of all supernodes. This completes the construction of G . Overall, $|V(G)| = O(n^5)$.

By “visiting” a supernode or the hypernode, we mean visiting any of their constituent nodes. We now make an immediate observation which allows us to restrict ourselves to $s - t$ paths P not visiting any supernode or the hypernode, i.e., with $V(P) \subseteq \mathcal{V}$.

Claim 2.1 *No $\omega(1/n)$ -approximate secluded path solution will visit any supernodes or the hypernode, i.e., will satisfy $V(P) \cap \mathcal{C} = \emptyset$.*

Proof: For every path P that goes through a node $v \in \mathcal{C} \cup \mathcal{H}$ it holds that $\text{Cost}(P) \geq n^5$. On the other hand, a path P' that visits only \mathcal{V} nodes, i.e., $V(P') \subseteq \mathcal{V}$ costs only $\text{Cost}(P') \leq X + |R| \cdot n^3 = O(n^4)$. The claim follows. ■

Let $\tilde{I}(P) = \{i \mid \mathcal{C}_i \subseteq N[P]\}$ be the indices of the supernodes in the neighborhood of path P . Then

$$\text{Cost}(P) = X + |\tilde{I}(P)| \cdot n^3.$$

We first show the correctness of the reduction and then consider gap-preservation.

Claim 2.2 *There exists an $s - t$ path of cost $X + kn^3$ iff there exists an **Red-Blue Set***

Cover solution of cost k .

Proof: (\Rightarrow) By Claim 2.1, we restrict ourselves to paths not visiting any supernodes. Thus, by the structure of the graph, any such $s - t$ path visits one of the nodes $v_{i,j}$ for each $1 \leq i \leq |B|$, corresponding to the selection of S_j to cover b_i . Since all X ordinary nodes will be in $N(P)$, kn^3 of the cost is due to the size- n^3 supernodes, which correspond to k red elements covered in the **Red-Blue Set Cover** solution.

(\Leftarrow) Conversely, given a cost- k **Red-Blue Set Cover** solution, we construct the path as follows. Starting from s , for every $1 \leq i \leq |B|$, add to P a node $v_{i,j}$ such that S_j is in the **Red-Blue Set Cover** solution, and then finally add t . The cost due to the ordinary nodes is X and due to the k supernodes is kn^3 . ■

We now show that the reduction is gap-preserving. By Claim 2.2 it follows that $q_{s,t}^* = X + |R^*| \cdot n^3$, where $|R^*|$ is the optimal **Red-Blue Set Cover** cost (the number of red elements R^* covered by the optimal solution). Assume that there exists an $\alpha > O(1/n)$ approximation algorithm for the **Secluded Steiner Tree** problem. This would result in path a P such that

$$\text{Cost}(P) = X + |\tilde{I}(P)| \cdot n^3 \leq \alpha(X + |R^*| \cdot n^3) \leq 2\alpha \cdot |R^*| \cdot n^3 .$$

Thus $|\tilde{I}(P)| \leq 2\alpha|R^*|$, implying that $(\text{Cost}(P) - X)/n^3$ is a 2α approximation to **Red-Blue Set Cover**. Since $|V(G)| = n^5$, and as **Red-Blue Set Cover** is inapproximable within a factor of $O(2^{\log^{1-\epsilon} n})$ for every fixed ϵ , we get that **Secluded Path** is inapproximable within a factor of $O(2^{\log^{1-\epsilon'} n})$, where $\epsilon' = f(\epsilon)$ for some appropriate function f . This complete the proof of Thm. 2.1 ■

Corollary 2.2 *The Secluded Path problem (and hence also the Secluded Steiner Tree problem) is strongly inapproximable in directed acyclic graphs.*

Proof: The proof is by gap-preserving reduction from **Red-Blue Set Cover**, very similar to the general case. The key modifications from the general case are that the hypernode of n^5 nodes is not included and we add directions to the edges; directing the edges from s toward the nodes of S_1 , namely, \mathcal{V}_1 , from the nodes of \mathcal{V}_i to the nodes of \mathcal{V}_{i+1} for every $i \in \{1, \dots, |B|\}$, and from the nodes of $\mathcal{V}_{|B|}$ to the node t . We also direct all the supernodes edges towards the supernode elements. It can be verified that this directed graph is acyclic. The rest of the analysis is as in the undirected case. The graph G contains $O(n^3)$ nodes, thus the **Secluded Path** problem cannot be approximated within an $O(2^{\log^{1-\epsilon} n})$ ratio. The corollary follows. ■

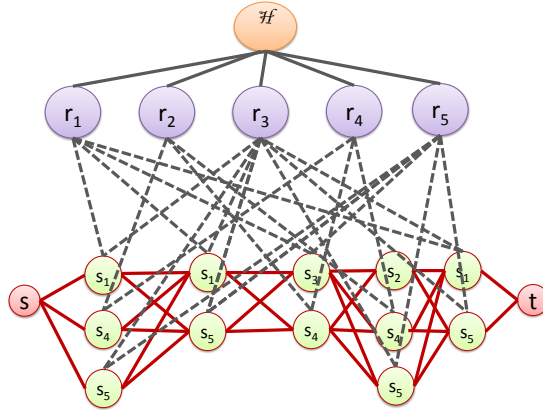


Figure 1: Example of the gap-preserving reduction from Red-Blue Set Cover to Secluded Path. The Red-Blue Set Cover instance is given by $S_1 = \{b_1, b_2, b_3, r_1, r_3\}$, $S_2 = \{b_4, r_4\}$, $S_3 = \{b_3, r_1, r_5\}$, $S_4 = \{b_1, b_3, b_4, r_2, r_4\}$, $S_5 = \{b_1, b_2, b_4, b_5, r_3, r_5\}$. The optimal solution for Red-Blue Set Cover is $\{S_1, S_3, S_5\}$, which covers 3 red elements. The corresponding $s - t$ path is of cost $14 + 3 \cdot 5^3 = 389$. Note that an alternative solution of $\{S_1, S_4\}$ is of higher cost, covering 4 red elements, and the corresponding $s - t$ path is of cost $14 + 4 \cdot 5^3 = 514$.

2.1 Approximation

Theorem 2.3 *The Secluded Path problem in unweighted undirected graphs can be approximated within a ratio of $\sqrt{\Delta} + 3$.*

Proof: Given an instance of the Secluded Path problem, let P^* be an $s - t$ path that minimizes $\text{Cost}(P^*)$. Note that we may assume without loss of generality that for every node u in P^* , the only neighbors of u in G from among the nodes of $V(P^*)$ are the nodes adjacent to u in P^* . To see this, note that otherwise, if u had an edges to some neighbor $u' \in V(P^*)$ such that e is not on P^* , we could have shortened the path P^* (by replacing the subpath from u to u' with the edge e) and obtained a shorter path with at most the same cost as P^* .

Recall that $\text{DegCost}(P)$ denotes the sum of the node degrees of the path P , and that $\text{Cost}(P) \leq \text{DegCost}(P)$ for any P . Recall also that the problem of finding an $s - t$ path P with minimum $\text{DegCost}(P)$ is polynomial. We claim that the algorithm that returns the path Q^* minimizing $\text{DegCost}(Q^*)$ yields a $(\sqrt{\Delta} + 3)$ approximation ratio for the Secluded Path problem.

In order to prove this, we show there exists a path Q such that $\text{DegCost}(Q) \leq (\sqrt{\Delta} + 3) \cdot \text{Cost}(P^*)$. This implies that $\text{DegCost}(Q^*) \leq \text{DegCost}(Q) \leq (\sqrt{\Delta} + 3) \cdot \text{Cost}(P^*)$, as required.

The path Q is constructed by the following iterative process. Initially, all nodes are unmarked and we set $Q = P^*$. While there exists a node with more than $\sqrt{\Delta} + 3$ unmarked neighbors on the path Q , pick such a node x . Let y be the first (closest to s) neighbor of x in Q and let z be the last (closest to t) neighbor of x in Q . Replace the subpath $Q[y, z]$ in Q with the path $[y, x, z]$ and mark the node x . We now show that $\text{DegCost}(Q) \leq (\sqrt{\Delta} + 3) \cdot \text{Cost}(P^*)$. Let $X = \{x_1, \dots, x_\ell\}$ be the set of marked nodes in Q , where x_i is the node marked in iteration i . Note that $\ell = |X|$ is the number of iterations in the entire process.

For the sake of analysis, partition $N[P^*]$ into two sets: S_1 , those that have no unmarked neighbor in Q , and S_2 , those that do. We claim that $|S_1| \geq \ell \cdot \sqrt{\Delta}$.

For each $x_i \in X$, let $P(x_i)$ be the path that was replaced by the process of constructing Q in iteration i , and let $Q(x_i)$ be the path Q in the beginning of that iteration. Since x_i has more than $\sqrt{\Delta} + 3$ unmarked neighbors in $Q(x_i)$, we get that $|P(x_i) \cap P^*| \geq \sqrt{\Delta} + 4$. Let $P^-(x_i)$ be the path obtained by removing the first two nodes and last two nodes from $P(x_i)$. Note that $|P^-(x_i) \cap P^*| \geq \sqrt{\Delta}$. In addition, the sets $P^-(x_i) \cap P^*$ for $i = \{1, \dots, \ell\}$ are pairwise disjoint, and moreover, none of the nodes in Q has a neighbor in $P^-(x_i) \cap P^*$. We thus get that $P^-(x_i) \cap P^* \subseteq S_1$. Note that $\text{Cost}(P^*) = |S_1| + |S_2|$ and that $\ell \leq |S_1| / \sqrt{\Delta}$. We get that $\text{DegCost}(Q) \leq |S_2| \cdot (\sqrt{\Delta} + 3) + \ell \cdot \Delta \leq |S_2| \cdot (\sqrt{\Delta} + 3) + |S_1| \cdot \sqrt{\Delta} \leq (|S_2| + |S_1|) \cdot (\sqrt{\Delta} + 3) = \text{Cost}(P^*) \cdot (\sqrt{\Delta} + 3)$. ■

In addition, for the **Secluded Steiner Tree** problem with k terminals, we establish the following.

Theorem 2.4 *The Secluded Steiner Tree problem in unweighted undirected graphs can be approximated within a ratio of $\min\{\Delta, n/k, \sqrt{2n}, O(\log k \cdot (k + \sqrt{\Delta}))\}$.*

Recall that $\text{DegCost}(T)$ is the sum of degrees in the tree T and let $\text{DegCost}^*(\mathcal{S}) = \min\{\text{DegCost}(T) \mid T \subseteq G \text{ spans } \mathcal{S}\}$. Clearly, $\text{Cost}(T) \leq \text{DegCost}(T)$.

The **Min Degree Steiner Tree** problem is defined as follows. Given a graph $G = (V, E)$ and a set of terminals \mathcal{S} , the objective is to find a Steiner tree T of minimal $\text{DegCost}(T)$, i.e., such that $\text{DegCost}(T) = \text{DegCost}^*(\mathcal{S})$.

Claim 2.3 *The Min Degree Steiner Tree problem is NP-hard for arbitrary $k = |\mathcal{S}|$*

and can be approximated within a ratio of $\Theta(\log k)$. If k is constant, then **Min Degree Steiner Tree** is polynomial.

Proof: We first present an $O(\log k)$ approximation algorithm. In the **Node Weighted Steiner Tree** problem, nonnegative costs are assigned to nodes as well as to edges. The cost of subgraph $G' \subseteq G$ that contains the terminals \mathcal{S} is the sum of the costs of its nodes and edges. While constant factor approximations exist for the standard **Steiner Tree** problem (of edge-weighted graphs), the **Node Weighted Steiner Tree** problem cannot be approximated to within less than a logarithmic factor $O(\log k)$ assuming that $NP \not\subseteq DTIME(n^{O(\log \log n)})$, by a reduction from **Set Cover** [8, 14]. Using the $O(\log k)$ -approximation algorithm of [8], the **Min Degree Steiner Tree** problem can be solved as follows. Set the weight of each vertex to be its degree, that is $W(v) = |N(v)|$, for every $v \in V(G)$, and set zero weights to the edges. The optimal **Node Weighted Steiner Tree** tree with these weights corresponds to the optimal **Min Degree Steiner Tree** tree. Since **Node Weighted Steiner Tree** is polynomial for fixed k , so is **Min Degree Steiner Tree** [14].

We now show that this approximation is tight. For brevity, let Π denote the special case of the **Node Weighted Steiner Tree** problem in which edges have zero weights and Π' denote the **Min Degree Steiner Tree** task. The approximation-preserving reduction from **Set Cover** presented in [14] assigns zero weights to the edges, and hence Π is inapproximable within logarithmic factor. By reducing from Π to Π' we will show that Π' is inapproximable within logarithmic factor as well. Given a Π instance graph $G(V, E, W)$ and terminal set \mathcal{S} , where $W(v_i) \geq 0$, we transform it into an instance G', \mathcal{S} of Π' . We assume that the weights are polynomial. To obtain G' , the weights W are scaled (multiplied by a common large enough factor X) to weights W' such that $W'(v_i) \geq |N_G(v_i)|$ for every node $v_i \in V$. Next, define the residual of vertex v_i as $Res(v_i) = W'(v_i) - |N_G(v_i)|$. The scaling step guarantees that $Res(v_i) \geq 0$. Next, add to each vertex v_i a disjoint set $\Upsilon(v_i) = \{v_i^j, j \in [1, Res(v_i)]\}$ consisting of 1-weight $Res(v_i)$ neighbors. Formally, $G' = (V', E', W')$, where $V' = V \cup_{v_i \in V} \Upsilon(v_i)$ and $E' = E \cup \{(v_i, v_i^j), v_i^j \in \Upsilon(v_i), v_i \in V\}$. Finally, $W'(v_i) = |N_{G'}(v_i)|$ for every $v_i \in V$ and $W'(u) = 1$ for every $u \in V' \setminus V$. It is easy to see that optimal solution of Π' in G' corresponds to an optimal solution for Π in G . Since the weights W are polynomial (hence so is X) it follows that a $\Omega(\log k)$ approximation for Π' in G' would imply an $\Omega(\log k)$ approximation for Π in G , in contradiction to the logarithmic inapproximability of Π' due to [14]. ■

We now show the following.

Lemma 2.5 $\text{DegCost}^*(\mathcal{S}) = \Theta(\sqrt{\Delta} + k) \cdot q^*(\mathcal{S})$.

Proof: Let T^* be an optimal secluded Steiner tree, meaning $\text{Cost}(T^*) = q^*(\mathcal{S})$. Starting with T^* we will perform a set of operations on it, ending with a spanning \mathcal{S} tree \widehat{T} for which $\text{DegCost}(\widehat{T}) \leq \Theta(\sqrt{\Delta} + k)\text{Cost}(T^*)$. Since $\text{DegCost}(\widehat{T}) \geq \text{DegCost}^*(\mathcal{S})$ this would establish that $\text{DegCost}^*(\mathcal{S}) < O(\sqrt{\Delta} + k) \cdot q^*(\mathcal{S})$. We begin by describing the procedure for obtaining \widehat{T} from T^* and then analyze its cost, i.e., $\text{DegCost}(\widehat{T})$. Note that as we do not know T^* ; this procedure is only theoretical to aid the analysis and is not meant for implementation.

Let $\Delta^* = \sqrt{\Delta} + 4k$. Throughout, we assume that $\Delta > \Delta^*$. Else, any $O(1)$ -approximation algorithm for the **Steiner Tree** problem is a $O(\Delta)$ -approximation algorithm to the **Secluded Steiner Tree** problem. Initially, let all nodes in T^* be unmarked and set $\widehat{T} = T^*$. A node u is *high-degree with respect to \widehat{T}* if it has at least Δ^* unmarked neighbors in \widehat{T} . While there is a high-degree node x with respect to \widehat{T} , do the following. For each subpath $\alpha = [a, b] \in \Sigma(T^*)$ such that $|N(x) \cap V(\alpha)| \geq 2$, i.e., x has at least two neighbors in α , replace the following subpaths. Let y_1 (respectively, y_2) be the neighbor of x closest to a (respectively, b) on T^* . Replace the subpath $[y_1, y_2] \subseteq \alpha$ with the subpath $[y_1, x, y_2]$ and mark the node x .

We now show that $\text{DegCost}(\widehat{T}) \leq O(\sqrt{\Delta} + k)q^*(\mathcal{S})$. Let $M = \{u_1, \dots, u_y\}$ be the set of marked nodes in \widehat{T} . This implies that the procedure consists of $|M| = y$ iterations. Let $T_0 = T^*$ and T_i be the spanning \mathcal{S} subgraph at the end of iteration i for $i = 1, \dots, |M|$ (thus $T_y = \widehat{T}$). The node $u_i \in M$ is the high-degree node with respect to T_{i-1} that was observed at iteration i . Let $\overline{M} = V(\widehat{T}) \setminus M$ be the unmarked nodes in the final subgraph \widehat{T} . Note that \widehat{T} might not be a tree (due to cycle introduced by the marked nodes), but clearly, a subgraph of it is a tree spanning all terminals. By definition, the DegCost of \widehat{T} is given by

$$\text{DegCost}(\widehat{T}) = \text{DegCost}(M) + \text{DegCost}(\overline{M}), \quad (3)$$

where $\text{DegCost}(U)$, for $U \subseteq V$, is given by $\text{DegCost}(U) = \sum_{u \in U} |N_G(u)|$. Let $M_i = \{u_j \mid j \leq i\}$ be the set of marked nodes at the end of iteration i . Given a path $P = [x_1, x_2, \dots, x_{q-1}, x_q]$, let $P^- = [x_2, \dots, x_{q-1}]$ denote its internal subpath. For every marked node u_i and a subpath $\alpha \in \Sigma(T^*)$ for which u_i has at least 3 unmarked neighbors in α at iteration i , let $P_i(\alpha)$ be the subpath in T_{i-1} that was shortcut in T_i by introducing u_i ; otherwise let $P_i(\alpha) = \emptyset$. Let $N^-(u_i) = N_{T_{i-1}}(u_i) \setminus M_{i-1}$ be the unmarked neighbors of u_i at iteration i and define $\Psi(u_i) = \bigcup_{\alpha \in \Sigma(T^*)} N^-(u_i) \cap V(P_i^-(\alpha))$, the set of “internal” unmarked neighbors of u_i in T_{i-1} (the nodes that are removed from T_{i-1} due to path

replacements).

We now make two observations. (A) $|\Psi(u_i)| \geq \sqrt{\Delta}$ for every node $u_i \in M$, and (B) $\Psi(u_1), \dots, \Psi(u_y)$ are pairwise disjoint. We start with (A). Note that for every $\alpha \in \Sigma(T^*)$ on which u_i has unmarked neighbors, we ignore at most 2 nodes in $\Psi(u_i)$ (corresponding to the endpoints of the replaced path, $P_i(\alpha)$). Since there are at most $2k$ subpaths in $\Sigma(T^*)$, there are at most $4k$ unmarked neighbors of u in T_{i-1} that are ignored in $\Psi(u_i)$. As u_i has Δ^* unmarked neighbors in T_{i-1} , we get that $|\Psi(u_i)| \geq \Delta^* - 4k \geq \sqrt{\Delta}$, which establishes (A). Note that (B) holds since each set $\Psi(u_i) \subseteq V(T_{i-1})$ is part of the tree in iteration i but is not part of the tree in iteration $i + 1$, i.e. $\Psi(u_i) \not\subseteq V(T_i)$ (due to the subpath replacement step). We now refer to Eq. (3) and bound $\text{DegCost}(M)$ and $\text{DegCost}(\overline{M})$. First, observe that $\text{DegCost}(M) \leq \sqrt{\Delta} \cdot q^*(\mathcal{S})$. This holds since every node $u_i \in M$ corresponds to a disjoint set $\Psi(u_i) \subseteq V(T^*)$ of at least $\sqrt{\Delta}$ nodes of the optimal tree T^* . Thus, $|M| \leq V(T^*)/\sqrt{\Delta}$ and $|M| \leq q^*(\mathcal{S})/\sqrt{\Delta}$. Since the degree of each marked node is bounded by Δ , it follows that

$$\text{DegCost}(M) \leq \sqrt{\Delta} \cdot q^*(\mathcal{S}). \quad (4)$$

We now bound the degree of the unmarked nodes \widehat{M} of \widehat{T} . Let $N^* = N_G[\widehat{M}]$. Since $\widehat{M} \subseteq V(T^*)$, it follows that $N^* \subseteq N[T^*]$ and hence $|N^*| \leq q^*(\mathcal{S})$. By the stopping criteria, it follows that any node in N^* has at most Δ^* neighbors in \widehat{M} . Hence

$$\text{DegCost}(\widehat{M}) \leq \Delta^* \cdot q^*(\mathcal{S}). \quad (5)$$

Combining Eqs. (3), (4), and (5), it follows that $\text{DegCost}^*(\mathcal{S}) = O(\sqrt{\Delta} + k) \cdot q^*(\mathcal{S})$. Finally, we show that there exists a graph G and a set of terminals \mathcal{S} for which $\text{DegCost}^*(\mathcal{S}) = \Omega(\sqrt{\Delta} + k) \cdot q^*(\mathcal{S})$. Set $3 < \Delta < k$, graph G consists of $2k + \Delta - 5$ nodes. Let $\mathcal{S} = \{v_0, \dots, v_{k-1}\}$. The skeleton for graph G is a path $P = [v_0, u_1, \dots, u_{k-2}, v_{k-1}]$. Then, for every $i \in [1, k - 2]$, connect u_i to the terminal v_i . In addition, connect each of the u_i nodes to the set of nodes $Z = \{z_1, \dots, z_{\Delta-3}\}$. The graph G is presented in Fig. 2. Clearly, the optimal tree T^* under both measures, i.e., $\text{Cost}(T^*)$ and $\text{DegCost}(T^*)$, is given by the subgraph induced by $G(V \setminus Z)$. Since every Z node is counted $k - 2$ times in $\text{DegCost}(T^*)$ instead of one time in $\text{Cost}(T^*)$, it follows that $\text{DegCost}(T^*) = \Delta \cdot (k - 2) + k$ and $\text{Cost}(T^*) = 2k + \Delta - 5$. Since $k \geq \Delta$, we get that $\text{DegCost}(T^*) = \Omega(\sqrt{\Delta} + k)\text{Cost}(T^*)$. The lemma follows. ■

Claim 2.4 *The Secluded Steiner Tree problem on unweighted undirected graphs can be approximated within $\sqrt{2n}$.*

Proof: Let g be a guess for $q^*(\mathcal{S})$, the solution of the optimal secluded Steiner tree. A Steiner tree T_g is constructed for each guess $g \in \{1, \dots, n\}$ as follows. Remove from G all vertices with degree $\geq g$ and let T_g be a 2-approximation for the optimal Steiner tree (with minimum number of edges). Let $T' = T_{g^*}$ for $g^* = q^*(\mathcal{S})$, i.e., the 2-approximation Steiner tree (where the objective is to minimize the number of edges) computed when considering the correct guess. We show that $\text{Cost}(T') \leq \sqrt{2n} \cdot q^*(\mathcal{S})$. Let L be the cost of the optimal Steiner tree (i.e., number of edges in the tree). Then clearly $q^*(\mathcal{S}) \geq L$. In addition, $\text{Cost}(T') \leq 2L \cdot q^*(\mathcal{S})$. This follows, since our approximated Steiner tree T' contains at most twice the edges of the optimal one, and the degree of each vertex is bounded by $q^*(\mathcal{S})$. Finally, since the cost of any secluded Steiner tree is bounded by the number of nodes n , we get that $\text{Cost}(T')/q^*(\mathcal{S}) = \min\{n/q^*(\mathcal{S}), 2 \cdot q^*(\mathcal{S})\} \leq \sqrt{2n}$. ■

We are now ready to complete the proof for Thm. 2.3.

Proof: First note that any $O(1)$ -approximation algorithm to the **Steiner Tree** problem is an $O(\Delta)$ -approximation algorithm to **Secluded Steiner Tree**. Next, note that obtaining an n/k approximation is trivial, since $\text{Cost}(T) \leq n$ for any spanning tree T and $q^*(\mathcal{S}) \geq k$. The approximation of $\sqrt{2n}$ is thanks to Claim 2.4 and that of $\Theta(\log k(\sqrt{\Delta} + k))$ is due to Claims 2.5 and 2.3. Specifically, let \mathcal{A} be an $\Theta(\log k)$ approximation algorithm for the **Min Degree Steiner Tree** problem, and let T be the resulting spanning \mathcal{S} tree output by algorithm \mathcal{A} given graph G and \mathcal{S} as input. Due to Claim 2.3, such an \mathcal{A} exists, and hence $\text{DegCost}(T) < \Theta(\log k)\text{DegCost}^*(\mathcal{S})$. Combining this with 2.5, we get that $\text{DegCost}(T) \leq \Theta(\log k(\sqrt{\Delta} + k)) \cdot q^*(\mathcal{S})$, as required. The theorem follows. ■

3 Bounded-Degree Graphs

In this section we show that the **Secluded Path** problem (and thus also the **Secluded Steiner Tree** problem) is *NP*-hard on both directed graphs and weighted graphs, even if the maximum node degree is 4. We have the following.

Lemma 3.1 *The Secluded Path problem is NP-complete even for graphs of maximum degree 4 if they are either (a) node-weighted, or (b) directed.*

We start with (a). We present a reduction from **Vertex Cover** of the weighted **Secluded Steiner Tree** problem. Throughout we consider the special case of the weighted **Secluded Steiner Tree** problem, where $\mathcal{S} = \{s, t\}$, namely, the **Secluded Path** problem. In [10]

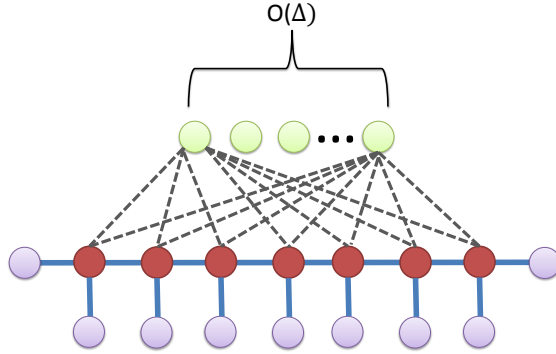


Figure 2: Illustration of the gap between $q^*(\mathcal{S})$ and $\text{DegCost}^*(\mathcal{S})$. The light purple nodes correspond to the set of terminals \mathcal{S} .

it was shown that the **Vertex Cover** problem is NP-complete even for planar graphs with maximum degree 3. Given a **Vertex Cover** instance in the form of a bounded-degree graph $G(V, E)$, the instance $(G'(V', E', W), s, t)$ to the weighted **Secluded Path** is constructed as follows. We add a “heavy” weight neighbor \hat{v} to every node $v \in V(G)$. In addition, every edge $e \in E$ is replaced by a gadget $g(e)$ consisting of four nodes, $V(g(e)) = \{U(e), D(e), L(e), R(e)\}$ forming a diamond graph. Let $S = \bigcup_{e \in E} V(g(e))$ and $\hat{V} = \bigcup_{v \in V} \{\hat{v}\}$. The set of nodes of G' is given by $V' = V \cup \hat{V} \cup S \cup \{s, t\}$. The weight function $W : V \rightarrow \mathbb{R}_{>0}$ is defined as follows: $W(v) = 1$ if $v \in V' \setminus \hat{V}$, and $W(v) = |V(G)| + 4|E|$ otherwise.

We now describe the gadget construction. Each edge $e = (u, v)$ is replaced by a diamond graph $g(e)$ consisting of the edges

$$E(g(e)) = \{(u, L(e)), (R(e), v), (L(e), U(e)), (L(e), D(e)), (U(e), R(e)), (D(e), R(e))\}$$

used to connect u and v in G' . (See Fig 3(a).) In addition, the new node s (respectively, t) is connected to the node $U(e')$ (respectively, $D(e'')$) for some arbitrary edges e', e'' , for $e' \neq e''$. Finally, fix some ordering e_1, \dots, e_m on the edges of E that starts with $e_1 = e'$ and ends with $e_m = e''$. Now, for every $i \in \{1, \dots, m-1\}$ connect $D(e_i)$ with $U(e_{i+1})$, creating a “tour” from s to t via the gadgets. This completes the description of the instance $(G'(V', E', W), s, t)$. For a path P and set of nodes W , denote the set of neighbors of P 's nodes in W by $N_W[P] = N[P] \cap W$. We now show that $N_{V(G)}[P]$ is an optimal vertex cover for G iff P is an optimal $s-t$ secluded path in the weighted graph G' . We begin by making an immediate observation.

Observation 3.2 For every optimal $s - t$ secluded path P , $V(P) \cap V(G) = \emptyset$.

Proof: Assume the contrary, and let P be an optimal secluded path with at least one node $v \in V(G)$. Since v has a heavy neighbor \hat{v} , it holds that $\text{Cost}(P) \geq 4E + V + 1$. Consider next an alternative path P' that goes strictly through the nodes in S , i.e., follows the tour through the gadgets $g(e)$. Since $N[P'] \subseteq \bigcup_{w \in S} N[w] \subseteq S \cup V(G)$, we get that $\text{Cost}(P) > 4E + V \geq \text{Cost}(P')$, in contradiction to the optimality of P . ■

Consequently, in the rest of the proof, we consider only secluded paths P that do not visit nodes of $V(G)$. Note that every such $s - t$ path is of length $3|E|$ and the set of S nodes is fully contained in neighbors of this path, i.e., $N_S[P] = S$. It therefore holds that,

$$\text{Cost}(P) = |S| + |N_{V(G)}[P]| . \quad (6)$$

The main factor determining the quality of an efficient path, among all paths that do not visit $V(G)$, is the number of path neighbors in $V(G)$, i.e., $|N_{V(G)}[P]|$. That is, by choosing between the u -side (respectively, v -side) in every gadget $g(e)$, for $e = (u, v)$, the resulting path has u (respectively, v) in its neighborhood, i.e., $N_{V(G)}[P]$. Overall, there are $|E|$ such decisions to be made. Let $VC(G)$ be an optimal vertex cover for G . We show the following.

Claim 3.1 $q_{s,t}^* = 4|E| + |VC(G)|$.

Proof: We begin by showing that $\text{Cost}(P) \geq 4|E| + |VC(G)|$ for every $s - t$ path P that does not go through the nodes of $V(G)$. Let $N^* = N_{V(G)}(P)$. We now show that N^* is a legal vertex cover of G , hence $|VC(G)| \leq |N^*|$ and by Eq. (6), $\text{Cost}(P) \geq 4|E| + |VC(G)|$ as required. Assume for contradiction that there exists an edge $e = (u, v) \in E$ not covered by the nodes of N^* , i.e., both $u, v \notin N^*$. Consider the gadget $g(e)$. Since P does not go through the nodes of $V(G)$, it passes through $U(e)$ and $D(e)$. In particular, when reaching $U(e)$, a decision is made between passing through the u side $L(e)$ or through the v side $R(e)$. Without loss of generality assume that P passes through $L(e)$, i.e., $P = [s, \dots, U(e), L(e), D(e), \dots, t]$. Then, since u is a neighbor of $L(e)$, it is in N^* and we end with contradiction.

Conversely, we now show that for every legal vertex cover C of G , there exists an $s - t$ secluded path P whose cost satisfies $\text{Cost}(P) \leq |C| + 4E$. The path P is constructed as follows. Start from s and traverse the gadgets $g(e)$ in order, from $g(e_1)$ to $g(e_m)$. Within each gadget $g(e)$, for $e = (u, v)$ such that $(u, L(e)) \in E'$ and $(v, R(e)) \in E'$, move from $U(e)$ to $D(e)$ through $L(e)$ if $u \in C$ and through $R(e)$ otherwise. One of these choices

must hold, since C is a legal vertex cover, so it must contain at least one of the nodes u and v . Hence, the path P “pays” for a node v in $V(G)$ only if $v \in C$. Formally, we get that $N^* \subseteq C$. Hence, combining with Eq. (6), $\text{Cost}(P) \leq 4E + |C|$. The claim follows. ■

Part (a) of the lemma follows. Finally, we now turn consider Lemma 3.1(b). It is sufficient to show a directed version for the reduction such that Obs. 3.2 still holds. Note that the main essence is to prevent any optimal path from visiting the nodes of $V(G)$ (and hence to visit every $g(e)$). In the weighted case, this goal is achieved by adding a “heavy” unique neighbor to every node in $V(G)$ (the black nodes in Fig. 3(a)). In the directed case, this property is achieved by directing the edges E' in the following manner. Recall the ordering e_1, \dots, e_m on the edges E , which imposes a direction for each edge in our construction as follows. We include the arcs $(s, U(e_1)), (D(e_m), t), (U(e_i), L(e_i)), (U(e_i), R(e_i)), (L(e_i), D(e_i)), (R(e_i), D(e_i))$, and in addition, $(D(e_i), U(e_{i+1}))$ for every $i \in \{1, \dots, m-1\}$. Finally, the edges between $L(e_i)$ and $R(e_i)$ to the nodes of $V(G)$ are directed towards the nodes of $V(G)$ (see Fig. 3(b)). This forces any $s-t$ path in G' to go through strictly the nodes in S . Obs. 3.2 now holds for the directed case, and the rest follows as in the weighted case. The lemma follows. ■

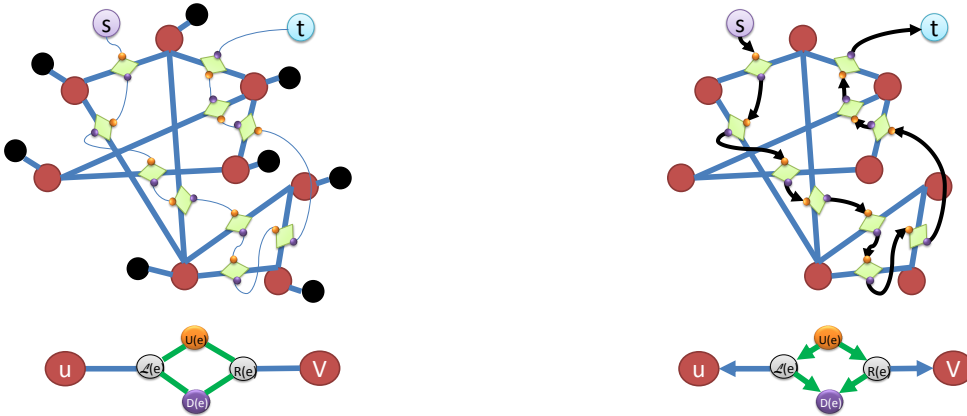


Figure 3: Illustration of the reduction from Vertex Cover. (a) The weighted case. Top: the graph G with the diamond graph gadgets $g(e)$. The black nodes correspond to the heavy neighbor attached to each node $v \in V(G)$. Bottom: zoom into a single gadget. (b) The directed case. Directionality enforces visiting the gadgets in order, precluding the tour in $V(G)$ nodes.

3.1 Polynomial-time algorithm for the unweighted undirected case.

In contrast, for unweighted undirected case of bounded-degree graphs, we show that the **Secluded Path** problem is polynomial. For two subpaths P_1, P_2 , define their asymmetric difference as

$$\text{Diff}(P_1, P_2) = |N[P_1] \setminus N[P_2]| .$$

Observation 3.3 (a) $\text{Diff}(P_1, P_2) \leq \text{Diff}(P_1, P')$ for every $P' \subseteq P_2$.

(b) $\text{Cost}(P_1 \circ P_2) = \text{Cost}(P_1) + \text{Diff}(P_2, P_1)$.

Proof: Part (a) simply follows as $N[P'] \subseteq N[P_2]$. Part (b) follows by definition. ■

We have the following.

Theorem 3.4 *The Secluded Path problem is polynomial on unweighted undirected degree-bounded graphs.*

Note that in the previous section we showed that if the graph is either weighted or directed then the **Secluded Path** problem (i.e., the special case of **Secluded Steiner Tree** with two terminals) is NP-hard. In addition, it is noteworthy that the related problem of **Minimum Labeled Path** problem [23, 11] is NP-hard even for unweighted planar graph with max degree 4 (this follows from a straightforward reduction from **Vertex Cover**).

We begin with notation and couple of key observations in this context. For a given path P , let $d_P(s, u)$ be the distance in edges between s and u in the path P . Recall, that Δ is the maximum degree in graph G .

Observation 3.5 *Let u, v be two nodes in some optimal $s-t$ path P^* that share a common neighbor, i.e., $N(u) \cap N(v) \neq \emptyset$. Then $d_{P^*}(u, v) \leq \Delta + 1$.*

Proof: Assume for contradiction that there exists an optimal $s-t$ path P^* such that $N(u) \cap N(v) \neq \emptyset$ where $u = P^*[i]$ and $v = P^*[j]$, $1 < i < j$ for $(i-j) \geq \Delta + 2$. Recall that due to the optimality of P^* , for every node u in P^* , the only neighbors of u in G from among the nodes of $V(P^*)$ are the nodes adjacent to u in P^* (otherwise the path can be shortcut). Let $w \in N(u) \cap N(v)$ be the mutual neighbor of u and v and consider the alternative $s-t$ path \widehat{P} obtained from P^* by replacing the subpath $Q = P[i, \dots, j]$ by the subpath $P' = [u, w, v]$. Let $Q^- = P[i+2, \dots, j-2]$ be an length- ℓ' internal subpath

of Q where $\ell' \geq \Delta - 1$. Then since the degree of w is at most Δ , it follows that

$$\mathbf{Cost}(\widehat{P}) \leq N[V(P) \setminus V(Q^-)] + \Delta - 2, \quad (7)$$

where $\Delta - 2$ is an upper bound on the number w 's neighbors other than u and v . In addition, note that by the optimality of P^* it contains no shortcut and hence $V(Q^-) \cap (N[V(P) \setminus V(Q^-)]) = \emptyset$ (i.e., for node $P[i]$ the only neighbors on the path P are $P[i - 1]$ and $P[i + 1]$). Thus,

$$\begin{aligned} \mathbf{Cost}(P^*) &\geq N[V(P) \setminus V(Q^-)] + |V(Q^-)| \\ &\geq N[V(P) \setminus V(Q^-)] + \Delta - 1 > \mathbf{Cost}(\widehat{P}), \end{aligned}$$

where the last inequality follows from Eq. (7), contradicting the optimality of P^* . The observation follows. \blacksquare

In other words, the observation says that two vertices on the optimal path at distance $\Delta + 2$ (which is constant for bounded-degree graphs) or more have no common neighbors. This key observation is at the heart of our dynamic program, as it enables the necessary subproblem independence property. The difficulty is that this observation applies only to optimal paths, and so a delicate analysis is needed to justify why the dynamic program works. Note that the main difficulty of computing the optimal secluded path P^* is that the cost function $\mathbf{Cost}(P^*)$ is not a linear function of path's components as in the related $\mathbf{DegCost}$ measure. Instead, the residual cost of the i th vertex in the path *depends* on the neighborhood of the length- $(i - 1)$ prefix of P^* . This dependency implies that the secluded path computation cannot be simply decomposed into independent subtasks. However, in contrast to suboptimal $s - t$ paths, the dependency (due to mutual neighbors) between the components of an optimal path is *limited* by the maximum degree Δ of the graph. The *limited dependency* exhibited by any $s - t'$ optimal path facilitates the correctness of the dynamic programming approach. Essentially, in the dynamic program, entries that correspond to subsolution σ of *any* optimal $s - t'$ path enjoy the limited dependency, and hence the values computed for these entries correspond to the *exact* cost of an optimal path that starts with s and ends with σ . In contrast, entries of subpaths σ that do not participate in any optimal $s - t'$ path, correspond to an *upper bound* on the cost of some path P that starts with s and ends with σ . This is due to the fact that the value of the entry is computed under the limited dependency assumption, and thus does not take into account the possible double counting of mutual neighbors between *distant* vertices in the path. Therefore, the possibly “falsified” entries cannot compete with the exact values, which are guaranteed to be computed for the entries that hold the subpaths of the optimal path. This informal intuition is formalized below.

For a path P of length $\ell \geq \Delta + 1$, let $\text{Suff}(P) = \langle v_{\ell-\Delta}, \dots, v_\ell \rangle$ be the $(\Delta + 1)$ -suffix of P .

Lemma 3.6 *Let P^* be an optimal $s - t'$ path of length $\geq \Delta + 1$. Let $P^* = P_1 \circ P_2$ be some partition of P^* into two subpaths such that $|P_1| \geq \Delta + 1$. Then $\text{Diff}(P_2, P_1) = \text{Diff}(P_2, \text{Suff}(P_1))$.*

Proof: For ease of notation, let $\ell = |P_1|$, $P' = P_1[1, \dots, \ell - (\Delta + 1)]$, and $\sigma = \text{Suff}(P_1)$. By definition, $\text{Diff}(P_2, P_1) = |(N[P_2] \setminus N[\sigma]) \setminus N[P']|$. In the same manner, $\text{Diff}(P_2, \sigma) = |(N[P_2] \setminus N[\sigma])|$. Assume for contradiction that the lemma does, namely, $\text{Diff}(P_2, \sigma) \neq \text{Diff}(P_2, P' \circ \sigma)$. Then by Obs. 3.3(a) we have that $\text{Diff}(P_2, \sigma) > \text{Diff}(P_2, P' \circ \sigma)$. This implies that $N[P_2] \cap N[P'] \neq \emptyset$. Let $u \in N[P_2] \cap N[P']$. There are two cases to consider: (a) $u \in P'$, and (b) u has a neighbor v_1 in P' . We handle case (a) by further dividing it into two subcases: (a1) $u \in P_2$, i.e., u occurs at least twice in P^* , once in P' and once in P_2 , and (a2) u has a neighbor v_2 in P_2 . Note that in both subcases, there exists a shortcut of P^* , obtained in subcase (a1) by cutting the subpath between the two duplicates of u and in subcase (a2) by shortcutting from u to v_2 . This shortcut results in a strictly lower cost path, in contradiction to the optimality of P^* . We proceed with case (b). Let $v_2 \in P_2$ be such that $u \in N[v_2]$. If $v_2 = u$, then clearly the path can be shortcut by going from $v_1 \in P'$ directly to $u \in P_2$, resulting in a lower cost path, in contradiction to the optimality of P^* . If $v_2 \neq u$, then $d_{P^*}(v_1, v_2) \geq \Delta + 2$ (since $|\sigma| \geq \Delta + 1$) and $N[v_1] \cap N(v_2) \neq \emptyset$, which in contradiction to Obs. 3.5. The Lemma follows. ■

Corollary 3.7 *Let P^* be an optimal $s - t'$ path of length $\geq \Delta + 1$. Let $P^* = P_1 \circ P_2$ be some partition of P^* into two subpaths such that $|P_1| \geq \Delta + 1$. Then $\text{Cost}(P^*) = \text{Cost}(P_1) + \text{Diff}(P_2, \text{Suff}(P_1))$.*

For clarity of representation, we describe a polynomial algorithm for the **Secluded Path** problem, i.e., where $\mathcal{S} = \{s, t\}$ and in addition $\Delta \leq 3$, in which case $\text{Suff}(P) = \langle v_{\ell-3}, \dots, v_\ell \rangle$. The general case of $\Delta(G) = O(1)$ is immediate by the description for the special case of $\Delta = 3$. The case of **Secluded Steiner Tree** with a fixed number of terminals $|\mathcal{S}| = O(1)$ is described in Subsection 3.2.

The algorithm we present is based on dynamic programming. For each $4 \leq i \leq n$, and every length-4 subpath given by a quartet of nodes $\sigma \subseteq V(G)$, it computes a length- i path $\pi(\sigma, i)$ that starts with s and ends with σ (if such exists) and an upper bound $f(\sigma, i)$ on the cost of this path, $f(\sigma, i) \geq \text{Cost}(\pi(\sigma, i))$. These values are computed inductively, using the values previously computed for other σ 's and $i - 1$. In contrast to the general

framework of dynamic programming, the interpretation of the computed values $f(\sigma, i)$ and $\pi(\sigma, i)$, namely, the relation between the dynamic programming values $f(\sigma, i)$ and $\pi(\sigma, i)$ and some “optimal” counterparts is more involved. In general, for arbitrary σ and i , the path $\pi(\sigma, i)$ is not guaranteed to be optimal in any sense and neither is its corresponding value $f(\sigma, i)$ (as $f(\sigma, i) \geq \mathbf{Cost}(\pi(\sigma, i))$). However, quite interestingly, there is a subset of quartets for which a useful characterization of $f(\sigma, i)$ and $\pi(\sigma, i)$ can be established. Specifically, for every $4 \leq i \leq n$, there is a subclass of quartets Ψ_i^* , for which the computed values are in fact “optimal”, in the sense that for every $\sigma \in \Psi_i^*$, the length- i path $\pi(\sigma, i)$ is of minimal cost among all other length- i paths that start with s and end with σ . We call such a path a *semi-optimal* path, since it is optimal only restricted to the length and specific suffix requirements. It turns out that for the special class of quartets Ψ_i^* , every semi-optimal path of $\sigma \in \Psi_i^*$ is also a prefix of some optimal $s - t'$ path. This property allows one to apply Cor. 3.7, which constitutes the key ingredient in our technique. In particular, it allows us to establish that $f(\sigma, i) = \mathbf{Cost}(\pi(\sigma, i))$. This is sufficient for our purposes since the set $\bigcup_{i=1}^n \Psi_i^*$ contains *any* quartet that occurs in *some* optimal secluded $s - t$ path P^* . Specifically, for every $s - t$ optimal path P^* , it holds that the quartet $\text{Suff}(P^*) = P^*[i - 3, i]$ satisfies $\text{Suff}(P^*) \in \Psi_i^*$. The correctness of the dynamic programming is established by the fact that the values computed for quartets that occur in optimal paths in fact correspond to optimal values as required (despite the fact the these values are “useless” for other quartets).

3.1.1 The algorithm

If the shortest path between s and t is less than 3, then the optimal **Secluded Path** can be found by an exhaustive search, so we assume throughout that $d_G(s, t) \geq 3$. Let Ψ be the set of all length-4 subpaths σ in G . That is, for every $\sigma \in \Psi$, we have $V(\sigma) \subseteq V(G)$ and $(\sigma[i], \sigma[i + 1]) \in E(G)$ for every $i \in [1, 3]$. For $\sigma \in \Psi$, define the collection of *shifted successor* of G as $\text{Next}(\sigma) = \{\langle \sigma[2], \dots, \sigma[4], u \mid u \in (N[\sigma[4]] \setminus \sigma) \rangle\}$. For every pair (σ, i) , where $\sigma \in \Psi$ and $i \in \{1, \dots, n\}$, the algorithm computes a value $f(\sigma, i)$ and length- i path $\pi(\sigma, i)$ ending with σ (i.e., $\pi(\sigma, i)[i - 3, \dots, i] = \sigma$). These values are computed inductively. For $i = 4$, let $\pi(\sigma, 4) = \sigma$ and

$$f(\sigma, 4) = \begin{cases} \mathbf{Cost}(\sigma), & \text{if } \sigma[1] = s \\ \infty, & \text{otherwise} \end{cases}$$

Once the algorithm has computed $f(\sigma, j)$ for every $4 \leq j \leq i - 1$ and every $\sigma \in \Psi$, in step $i \in \{5, n\}$ it computes

$$f(\sigma, i) = \min\{f(\sigma', i - 1) + \text{Diff}(\sigma, \sigma') \mid \sigma' \in \text{Next}(\sigma')\}. \quad (8)$$

Note that

$$\text{Diff}(\sigma, \sigma') = \text{Diff}(\sigma[4], \sigma') = |N[\sigma[4]] \setminus N[\sigma']|. \quad (9)$$

Let $\sigma' \in \Psi$ such that $\sigma \in \text{Next}(\sigma')$ and σ' achieves the minimum value in Eq. (8). Then define $\text{Pred}(\sigma) = \sigma'$ and let

$$\pi(\sigma, i) = \pi(\text{Pred}(\sigma), i - 1) \circ \sigma[4]. \quad (10)$$

Let $q_{s,t}^* = \min\{f(\sigma, i) \mid i \in \{4, \dots, n\}, \text{Last}(\pi(\sigma, i)) = t\}$, and set $P^* = \pi(\sigma^*, i^*)$ for σ^*, i^* such that $f(\sigma^*, i^*) = q_{s,t}^*$. Note that there are at most $O(n^4 \cdot n)$ entries $f(\sigma, i)$, each computed in constant time, and so the overall running time is $O(n^5)$.

Analysis. Throughout this section we make use of two notions of optimality for secluded paths. The two notions are parameterized by $\sigma \in \Psi$ and by an index $i \in \{1, \dots, n\}$. Let $\mathcal{P}(\sigma, i)$ be the set of length- i paths P starting at $P[1] = s$ and ending with $\text{Suff}(P) = \sigma$ for $P \in \mathcal{P}(\sigma, i)$. Denote the minimum cost of a path in $\mathcal{P}(\sigma, i)$ by $q^*(\sigma, i) = \min\{\text{Cost}(P) \mid P \in \mathcal{P}(\sigma, i)\}$ and denote the set of paths attaining this minimum cost by $\mathcal{Q}_{\text{semi}}(\sigma, i) = \{P \in \mathcal{P}(\sigma, i) \mid \text{Cost}(P) = q^*(\sigma, i)\}$. We refer to paths in $\mathcal{Q}_{\text{semi}}(\sigma, i)$ as *semi-optimal* paths, since they are optimal only among length- i paths ending with σ . Note that in general, a semi-optimal path $P \in \mathcal{Q}_{\text{semi}}(\sigma, i)$ is not necessarily an optimal secluded path from s to $P[i]$, nor is it a prefix of an optimal secluded path from s to any t' , and thus Cor. 3.7 cannot be applied. However, we next argue that there exists a subset of quartets $\sigma^* \in \Psi$ for which every path $P \in \mathcal{Q}_{\text{semi}}(\sigma, i)$ is a prefix of an optimal $s - t'$ secluded path, and since Obs. 3.5 holds for any subpath of some optimal $s - t'$ path, we can apply it for this subset of sequences. Let $\mathcal{Q}_{t'}^* = \{P \in \mathcal{P}_{s,t'} \mid \text{Cost}(P) = q_{s,t'}^*\}$ be the set of all optimal $s - t'$ paths for $t' \in V(G) \setminus \{s\}$. A path P that is a *prefix* of some optimal $s - t'$ path is hereafter referred to as a *pref-optimal* path. Let $\mathcal{Q}^*(\sigma, i) = \{P \in \bigcup_{t' \in V(G) \setminus \{s\}} \mathcal{Q}_{t'}^* \mid P[i - 3, i] = \sigma\}$. That is, $\mathcal{Q}^*(\sigma, i)$ is the set of all optimal secluded paths in which the subpath in positions $(i - 3)$ to i is equal to σ . Note that in contrast to the set $\mathcal{Q}_{\text{semi}}(\sigma, i)$, which is nonempty for every $\sigma \in \Psi$, and might contain suboptimal secluded paths, the set $\mathcal{Q}^*(\sigma, i)$ might be empty for many choices of σ , but every path $P \in \mathcal{Q}^*(\sigma, i)$ is indeed optimal.

Define the subset of subpaths for which $\mathcal{Q}^*(\sigma, i)$ is nonempty, for $i \in \{4, \dots, n\}$, as

$$\Psi_i^* = \{\sigma \in \Psi \mid \mathcal{Q}^*(\sigma, i) \neq \emptyset\}.$$

We next establish some useful properties for $\sigma \in \Psi_i^*$. Let $\mathcal{Q}^T(\sigma, i)$ correspond to the set of paths in $\mathcal{Q}^*(\sigma, i)$ but truncated at position i , that is, $\mathcal{Q}^T(\sigma, i) = \{P[1, i] \mid P \in \mathcal{Q}^*(\sigma, i)\}$. Clearly, every path P in $\mathcal{Q}^T(\sigma, i)$ is of length i . Moreover, since $\mathcal{Q}^*(\sigma, i)$ contains optimal paths, $\mathcal{Q}^T(\sigma, i)$ consists of pref-optimal paths. It is important to understand the distinction between the set $\mathcal{Q}^T(\sigma, i)$ and the set $\mathcal{Q}_{semi}(\sigma, i)$. Both sets contain length- i paths that start with s and end with σ . However, while $P_1 \in \mathcal{Q}^T(\sigma, i)$ (if it exists) corresponds to a prefix of some $s - t'$ optimal path, a path $P_2 \in \mathcal{Q}_{semi}(\sigma, i)$ is not necessarily a prefix of some optimal path; it is only optimal among paths of length i ending with σ . In particular, this difference implies that Cor. 3.7 can be safely applied to paths in $\mathcal{Q}^T(\sigma, i)$.

We first provide a general observation regarding the computed values $f(\sigma, i)$ and $\pi(\sigma, i)$.

Observation 3.8 *For every $\sigma \in \Psi$ and $i \in \{1, \dots, n\}$, if $P^* \in \mathcal{Q}_{semi}(\sigma, i)$ then $\text{Cost}(P^*) \leq \text{Cost}(\pi(\sigma, i)) \leq f(\sigma, i)$.*

Proof: The left inequality follows immediately, since both $\pi(\sigma, i)$ and P^* are in $\mathcal{P}(\sigma, i)$ and P^* is a semi-optimal path, i.e., it has the lowest cost among these paths. The right inequality is proved by induction on i . For $i = 4$, the claim holds trivially. Assume it holds for every σ and every $j \leq i - 1$, and consider a fixed σ and i . Let $\sigma' = \text{Pred}(\sigma)$. Then

$$\begin{aligned} \text{Cost}(\pi(\sigma, i)) &= \text{Cost}(\pi(\sigma', i - 1) \circ \sigma[4]) \\ &= \text{Cost}(\pi(\sigma', i - 1)) + \text{Diff}(\sigma[4], \pi(\sigma', i - 1)) \end{aligned} \quad (11)$$

$$\leq f(\sigma', i - 1) + \text{Diff}(\sigma[4], \sigma') \quad (12)$$

$$= f(\sigma', i - 1) + \text{Diff}(\sigma, \sigma') \quad (13)$$

$$= f(\sigma, i), \quad (14)$$

where Ineq. (11) follows from Obs. 3.3(b), Ineq. (12) follows from the inductive assumption, Eq. (13) follows from Eq. (9) and Eq. (11) follows from Eq. (8). The observation follows. ■

It turns out that for the set of special quartets that occur on some optimal $s - t$ path, a stronger characterization of $\pi(\sigma, i)$ and $f(\sigma, i)$ can be established.

Lemma 3.9 *For every i and $\sigma \in \Psi_i^*$,*

(a) $\pi(\sigma, i) \in \mathcal{Q}_{semi}(\sigma, i)$, and

(b) $f(\sigma, i) = \text{Cost}(\pi(\sigma, i))$.

Let P^* be some optimal secluded path and let $\sigma = \text{Suff}(P^*)$. Since $\sigma \in \Psi_{|P^*|}^*$, by Lemma 3.9 we get that the computed path $\pi(\sigma, |P^*|)$ is indeed an optimal $s - t$ path of cost $f(\sigma, |P^*|)$. Our remaining goal is therefore to establish Lemma 3.9.

Proof Sketch. We prove that the inequalities of Obs. 3.8 are in fact *equalities* for every $4 \leq i \leq n$ and every $\sigma \in \Psi_i^*$. The proof, by induction on i , is not immediate and requires several properties to be established and to come into play together. To begin, assume that both parts of Lemma 3.9 hold for every $\sigma \in \Psi_i^*$ for every $4 \leq i \leq \ell - 1$ and consider some $\sigma \in \Psi_\ell^*$. In addition to the inductive assumption, we assume that $\text{Pred}(\sigma) \in \Psi_{\ell-1}^*$ (which would be established later on). Given these assumptions, for part (a) of the lemma we show that $\pi(\sigma, \ell) = \pi(\text{Pred}(\sigma), \ell - 1) \circ \sigma[4] \in \mathcal{Q}_{\text{semi}}(\sigma, \ell)$. Next, to show that part (b) also holds, i.e., that $f(\sigma, \ell) = \text{Cost}(\pi(\sigma, \ell))$, we would like to show that $\pi(\sigma, \ell)$ is a pref-optimal path and thus Cor. 3.7 can be applied. Our argumentation can be sketched as follows. We prove that for every $\sigma \in \Psi^*(\ell)$ it holds that (a) every semi-optimal path $P \in \mathcal{Q}_{\text{semi}}(\sigma, \ell)$ is pref-optimal, and (b) there is a set of quartets denoted by $\text{Parent}(\sigma, \ell) \subseteq \Psi_{\ell-1}^*$, for which the following holds for every $\sigma' \in \text{Parent}(\sigma, \ell)$: (b1) $P' = P \circ \sigma[4] \in \mathcal{Q}_{\text{semi}}(\sigma', \ell)$ for every $P \in \mathcal{Q}_{\text{semi}}(\sigma', \ell - 1)$, and hence P' is pref-optimal, and (b2) $\text{Pred}(\sigma) \in \text{Parent}(\sigma, \ell)$ and thus the inductive hypothesis can be established.

Detailed analysis. We proceed by showing that every semi-optimal length- i path $P \in \mathcal{Q}_{\text{semi}}(\sigma, i)$ for $\sigma \in \Psi_i^*$, $i \in \{4, \dots, n\}$ is a pref-optimal path.

Lemma 3.10 *For every $\sigma \in \Psi_i^*$, $\mathcal{Q}^T(\sigma, i) = \mathcal{Q}_{\text{semi}}(\sigma, i)$.*

Proof: We first show that $\mathcal{Q}^T(\sigma, i) \subseteq \mathcal{Q}_{\text{semi}}(\sigma, i)$. To do that, we consider a path $P_1 \in \mathcal{Q}^*(\sigma, i)$ for which $P_1[1, i] \in \mathcal{Q}^T(\sigma, i)$, and show that $\text{Cost}(P_1[1, i]) \leq q^*(\sigma, i)$. Thus $P_1[1, i] \in \mathcal{Q}_{\text{semi}}(\sigma, i)$. Let $\ell = |P_1|$ and $t' = \text{Last}(P_1)$, thus P_1 is an optimal $s - t'$ path. Assume for contradiction that $\text{Cost}(P_1[1, i]) > q^*(\sigma, i)$. Let $P_2 \in \mathcal{Q}_{\text{semi}}(\sigma, i)$, i.e., $\text{Cost}(P_2) = q^*(\sigma, i)$. Consider an alternative $s - t'$ path $P_3 = P_2 \circ P_1[i + 1, \ell]$. Note that P_3 is indeed a legal $s - t'$ path since $P_2[i] = P_1[i]$. We now compute its cost and compare it to that of P_1 .

$$\text{Cost}(P_3) = \text{Cost}(P_2) + \text{Diff}(P_1[i + 1, \ell], P_2) \quad (15)$$

$$\leq \text{Cost}(P_2) + \text{Diff}(P_1[i + 1, \ell], \sigma) \quad (16)$$

$$< \text{Cost}(P_1[1, i]) + \text{Diff}(P_1[i + 1, \ell], \sigma) \quad (17)$$

$$= \text{Cost}(P_1), \quad (18)$$

where Ineq. (15) and (16) follow by parts (b) and (a) of Obs. 3.3 and Ineq. (17) is by the contradiction assumption. Eq. (18) follows from the fact that P_1 is an optimal $s - t'$ path and thus Cor. 3.7 can be safely applied. We therefore get that $\text{Cost}(P_3) < \text{Cost}(P_1)$, in contradiction to the optimality of P_1 .

Conversely, we now prove that $\mathcal{Q}_{\text{semi}}(\sigma, i) \subseteq \mathcal{Q}^T(\sigma, i)$. Let $P_1 \in \mathcal{Q}_{\text{semi}}(\sigma, i)$ and $P_2 \in \mathcal{Q}^*(\sigma, i)$. Let $\ell = |P_2|$ and $t' = \text{Last}(P_2)$. Consider the path $P_3 = P_1 \circ P_2[i + 1, \ell]$ obtained by replacing the length- i prefix of P_2 with P_1 . We show that P_3 has the same cost, $\text{Cost}(P_3) = \text{Cost}(P_2)$, hence P_3 is also an optimal $s - t'$ path. This would imply that $P_3 \in \mathcal{Q}^*(\sigma, i)$ and $P_1 \in \mathcal{Q}^T(\sigma, i)$, establishing the lemma.

$$\begin{aligned} \text{Cost}(P_3) &= \text{Cost}(P_1 \circ P_2[i + 1, \ell]) = \text{Cost}(P_1) + \text{Diff}(P_2[i + 1, \ell], P_1) \\ &\leq \text{Cost}(P_1) + \text{Diff}(P_2[i + 1, \ell], \sigma) \\ &\leq \text{Cost}(P_2[1, i]) + \text{Diff}(P_2[i + 1, \ell], \sigma) & (19) \\ &= \text{Cost}(P_2), & (20) \end{aligned}$$

where Ineq. (19) follows from the fact that both P_1 and $P_2[1, i]$ are in $\mathcal{P}(\sigma, i)$, i.e., both are length- i paths starting at s and ending with σ and P_1 is semi-optimal. Ineq. (20) follows from the fact that P_2 is an optimal $s - t'$ path, thus Cor. 3.7 can be safely applied. The fact that P_2 is optimal necessitates equality. The lemma follows. \blacksquare

For $\sigma \in \Psi_i^*$ and $i \in \{5, \dots, n\}$, define

$$\text{Parent}(\sigma, i) = \{\sigma' = \langle u, \sigma[1], \dots, \sigma[3] \rangle \mid \exists P \in \mathcal{Q}^*(\sigma, i), P[i - 4, i] = u \circ \sigma\}.$$

That is, $\sigma' \in \text{Parent}(\sigma, i)$ iff there exist some $t' \in V(G) \setminus \{s\}$ and an optimal $s - t'$ path P^* such that $P^*[i - 4, i] = \sigma'(1) \circ \sigma$. Note that by the definition of Ψ_i^* , there exists at least one pref-optimal path P of length i that ends with σ . In addition, since $i \geq 5$ there exists at least one $\sigma' \in \text{Parent}(\sigma, i)$ corresponding to $\sigma' = P[i - 4, i - 1]$. In other words, $\text{Parent}(\sigma, i) \neq \emptyset$ for every $i \geq 5$ and $\sigma \in \Psi_i^*$. Lemma 3.10 implies that the following holds for every $\sigma \in \Psi_i^*$.

Observation 3.11 *If $\sigma \in \Psi_i^*$ then*

- (a) *for every $\sigma' \in \text{Parent}(\sigma, i)$ there exists a semi-optimal path $P^* \in \mathcal{Q}_{\text{semi}}(\sigma, i)$, which is also a pref-optimal path, such that $P^*[i - 4, i] = \sigma' \circ \sigma[4]$.*
- (b) *$\text{Parent}(\sigma, i) \subseteq \Psi_{i-1}^*$.*

We first provide an auxiliary claim that holds for every $i \in \{1, \dots, n\}$. Let σ, σ' be such that (a) $\sigma \in \Psi_i^*$ and (b) $\sigma' \in \text{Parent}(\sigma, i)$.

Claim 3.2 For every $\widehat{P} \in \mathcal{Q}_{semi}(\sigma', i-1)$, $\widehat{P} \circ \sigma[4] \in \mathcal{Q}_{semi}(\sigma, i)$

Proof: Let $P \in \mathcal{Q}_{semi}(\sigma, i)$ such that $\sigma' = P[i-4, i-1]$. Note that since $\sigma' \in Parent(\sigma, i)$, by Obs. 3.11(a), such a P is guaranteed to exist. Let $\widehat{P} \in \mathcal{Q}_{semi}(\sigma', i-1)$ and let $P'' = \widehat{P} \circ \sigma[4]$. We now prove that $\mathbf{Cost}(P'') \leq \mathbf{Cost}(P)$ and therefore $P'' \in \mathcal{Q}_{semi}(\sigma, i)$.

$$\begin{aligned} \mathbf{Cost}(P'') &= \mathbf{Cost}(\widehat{P} \circ \sigma[4]) = \mathbf{Cost}(\widehat{P}) + \text{Diff}(\sigma[4], \widehat{P}) \\ &\leq \mathbf{Cost}(\widehat{P}) + \text{Diff}(\sigma[4], \sigma') \end{aligned} \tag{21}$$

$$\leq \mathbf{Cost}(P[1, i-1]) + \text{Diff}(\sigma[4], \sigma') \tag{22}$$

$$= \mathbf{Cost}(P) , \tag{23}$$

where Ineq. (21) follows from Obs. 3.3(a), Ineq. (22) follows from the fact that both $P[1, i-1], \widehat{P} \in \mathcal{P}(\sigma', i-1)$ and \widehat{P} , being semi-optimal, has the minimal cost in this set. Eq. (23) follows from the fact that $P \in \mathcal{Q}_{semi}(\sigma, i)$ and since $\sigma \in \Psi_i^*$, Lemma 3.10 implies that P is a pref-optimal, i.e., a prefix of some optimal secluded path, i.e., $P \in \mathcal{Q}^T(\sigma, i)$. Therefore, Cor. 3.7 can be applied. ■

We now turn to proving Lemma 3.9, showing that for every i , and every $\sigma \in \Psi_i^*$, the inequalities of Obs. 3.8 become equalities.

Proof of Lemma 3.9: The lemma is proven by induction on i . For $i = 4$, both claims clearly hold. Assume the claims hold for $i \leq \ell - 1$ and consider ℓ . Let $\sigma' = \text{Pred}(\sigma)$. To be able to apply the inductive hypothesis to σ' , we first show the following.

Claim 3.3 $\sigma' \in Parent(\sigma, \ell)$.

Proof: Note that since $\sigma \in \Psi_\ell^*$, it holds that $Parent(\sigma, \ell) \neq \emptyset$, and also $Parent(\sigma, \ell) \subseteq \Psi_{\ell-1}^*$ by Obs. 3.11(b). Let $\sigma^* \in Parent(\sigma, \ell)$, thus $\sigma^* \in \Psi_{\ell-1}^*$. By Lemma 3.10, $\mathcal{Q}^T(\sigma, \ell) = \mathcal{Q}_{semi}(\sigma, \ell)$ and $\mathcal{Q}^T(\sigma^*, \ell-1) = \mathcal{Q}_{semi}(\sigma^*, \ell-1)$. Let us pick three arbitrary representatives $P^*(\sigma^*, \ell-1) \in \mathcal{Q}_{semi}(\sigma^*, \ell-1)$, $P^*(\sigma', \ell-1) \in \mathcal{Q}_{semi}(\sigma', \ell-1)$, $P^*(\sigma, \ell) \in \mathcal{Q}_{semi}(\sigma, \ell)$. Recall that due to Lemma 3.10, $P^*(\sigma^*, \ell-1)$ and $P^*(\sigma, \ell)$ are pref-optimal paths and therefore Cor. 3.7 can be safely applied for them. We first show that

$$\mathbf{Cost}(P^*(\sigma, \ell)) = \mathbf{Cost}(P^*(\sigma^*, \ell-1)) + \text{Diff}(\sigma, \sigma^*). \tag{24}$$

To see this, let $P' = P^*(\sigma^*, \ell-1) \circ \sigma[4]$. Then by Claim 3.2, $P' \in \mathcal{Q}_{semi}(\sigma, \ell)$. In addition, by Lemma 3.10, $P' \in \mathcal{Q}^T(\sigma, \ell)$ and therefore Cor. 3.7 can be safely applied, so Eq. (24) is established. We next show that $P'' = P^*(\sigma', \ell-1) \circ \sigma[4] \in \mathcal{Q}_{semi}(\sigma, \ell)$. Clearly, $P'' \in \mathcal{P}(\sigma, \ell)$, so it suffices to show that $\mathbf{Cost}(P'') \leq \mathbf{Cost}(P^*(\sigma, \ell))$ as $P^*(\sigma, \ell)$ is

semi-optimal. Indeed,

$$\begin{aligned} \mathbf{Cost}(P'') &= \mathbf{Cost}(P^*(\sigma', \ell - 1)) + \text{Diff}(\sigma[4], P^*(\sigma', \ell - 1)) \\ &\leq \mathbf{Cost}(P^*(\sigma', \ell - 1)) + \text{Diff}(\sigma[4], \sigma') \end{aligned} \quad (25)$$

$$\leq f(\sigma', \ell - 1) + \text{Diff}(\sigma, \sigma') \quad (26)$$

$$\leq f(\sigma^*, \ell - 1) + \text{Diff}(\sigma, \sigma^*) \quad (27)$$

$$= \mathbf{Cost}(\pi(\sigma^*, \ell - 1)) + \text{Diff}(\sigma, \sigma^*) \quad (28)$$

$$= \mathbf{Cost}(P^*(\sigma^*, \ell - 1)) + \text{Diff}(\sigma, \sigma^*) = \mathbf{Cost}(P^*(\sigma, \ell)) \quad (29)$$

where Ineq. (25) follows from Obs. 3.3(a) and Ineq. (26) follows from Lemma 3.8 and Eq. (9). Ineq. (27) follows from the fact that $\sigma \in \text{Next}(\sigma') \cap \text{Next}(\sigma^*)$ and σ' attains the minimum value of Eq. (8). Eq. (28) follows from part (b) of the inductive assumption for $\ell - 1$. In Eq. (29) the first equality follows from the fact that both $\pi(\sigma^*, \ell - 1)$ and $P^*(\sigma^*, \ell - 1)$ are in $\mathcal{Q}_{\text{semi}}(\sigma^*, \ell - 1)$, and thus $\mathbf{Cost}(P^*(\sigma^*, \ell - 1)) = \mathbf{Cost}(\pi(\sigma^*, \ell - 1)) = q^*(\sigma^*, \ell - 1)$, and the second equality follows from Eq. (24).

We therefore get that $P'' \in \mathcal{Q}_{\text{semi}}(\sigma, \ell)$ and since $\sigma \in \Psi_\ell^*$, by Lemma 3.10 it holds that $P'' \in \mathcal{Q}^T(\sigma, \ell)$, i.e., $P'' = P^*(\sigma', \ell - 1) \circ \sigma[4]$ is also a pref-optimal path. This implies that $\sigma' \in \text{Parent}(\sigma, \ell)$, as required. ■

By Claim 3.3 and Obs. 3.11(b), $\sigma' \in \Psi_{\ell-1}^*$. It follows that the inductive hypothesis can be applied to σ' . By part (a) of the inductive assumption for $\ell - 1$, it holds that $\pi(\sigma', \ell - 1) \in \mathcal{Q}_{\text{semi}}(\sigma', \ell - 1)$. Finally, by Claim 3.2 we get that the path $\pi(\sigma, \ell)$ chosen by the algorithm satisfies $\pi(\sigma, \ell) = \pi(\sigma', \ell - 1) \circ \sigma[4] \in \mathcal{Q}_{\text{semi}}(\sigma, \ell)$, which establishes (a). We now consider (b). Note that

$$\begin{aligned} f(\sigma, \ell) &= f(\sigma', \ell - 1) + \text{Diff}(\sigma, \sigma') \\ &= \mathbf{Cost}(\pi(\sigma', \ell - 1)) + \text{Diff}(\sigma, \sigma') \end{aligned} \quad (30)$$

$$= \mathbf{Cost}(\pi(\sigma', \ell - 1) \circ \sigma[4]), \quad (31)$$

where Eq. (30) follows from the inductive hypothesis for $\ell - 1$ and Eq. (31) follows from the fact that $\pi(\sigma', \ell - 1) \circ \sigma[4] \in \mathcal{Q}^T(\sigma, \ell)$ and thus Cor. 3.7 can be applied. This completes the proof of Lemma 3.9. We are now ready to complete the correctness proof of the algorithm. Let P^* be an optimal $s - t$ path of length ℓ . Then $P^*[\ell - 3, \ell] \in \Psi_\ell^*$. Therefore, by Lemma 3.9, $P^*[1, \ell] \in \mathcal{Q}_{\text{semi}}(P^*[\ell - 3, \ell], \ell)$, and so $f(P^*[\ell - 3, \ell], \ell) = \mathbf{Cost}(P^*)$, and by Lemma 3.8, the cost of any other σ such that $t \in \sigma$, and every i , satisfies $f(\sigma, i) \geq \pi(\sigma, i) \geq t$. Thm. 3.4 is established.

Corollary 3.12 *The Secluded Path problem can be solved in $O(n^{\Delta+3})$ time.*

This follows since there are $O(n^{\Delta+1})$ distinct length- $(\Delta + 1)$ subpaths, and the length of the path is bounded by n . Hence, overall there are $O(n^{\Delta+2})$ entries and the value for each entry is computed in $O(\Delta)$ time. For the **Secluded Steiner Tree** problem, we show the following.

Theorem 3.13 *On unweighted undirected degree-bounded graphs, we have the following: (a) for arbitrary k , the **Secluded Steiner Tree** problem is NP-hard; (b) for $k = O(1)$, the **Secluded Steiner Tree** problem is polynomial.*

Proof: We start with (a). We prove this by reduction from **Vertex Cover**. By [10], we know that **Vertex Cover** is NP-complete even for planar graphs with maximum degree 3. Given an instance of the **Vertex Cover** problem consists of a bounded-degree graph $G = (V, E)$, we transform it into an instance of the **Secluded Steiner Tree** problem that consists of a bounded-degree graph \widehat{G} and a set of terminal \mathcal{S} . Let \widehat{B} be a complete binary tree with n leaves $L = \{\ell_1, \dots, \ell_n\}$. Consider some ordering on $V(G)$ and connect ℓ_i to v_i , for every $i \in \{1, \dots, n\}$. In addition, for each vertex $v_i \in V(G)$, add a single neighbor \widehat{v}_i in \widehat{G} . Next, for each edge $e = (v_i, v_j) \in E(G)$, add a vertex v'_e . Every edge $e \in E(G)$ is replaced by the two edges (v'_e, v_i) and (v'_e, v_j) in \widehat{G} . In sum, the graph \widehat{G} is defined by

$$V(\widehat{G}) = V(G) \cup V(\widehat{B}) \cup \bigcup_{v_i \in V(G)} \{\widehat{v}_i\} \cup \bigcup_{e \in E(G)} \{v'_e\},$$

and

$$E(\widehat{G}) = E(\widehat{B}) \cup \bigcup_{i=1}^n \{(\ell_i, v_i), (\widehat{v}_i, v_i)\} \cup \bigcup_{e \in E(G)} \{(v'_e, v_i), (v'_e, v_j)\}.$$

Note that the maximum degree $\Delta(\widehat{G})$ in \widehat{G} is at most $\Delta(G) + 1$. Hence \widehat{G} is a bounded-degree graph as well. The terminal set $\mathcal{S} = \bigcup_{e \in E(G)} \{v'_e\} \cup V(\widehat{B})$ includes the v'_e nodes and the nodes of the binary tree $V(\widehat{B})$. We now show that G has vertex cover of size k iff there exists a secluded Steiner tree T in \widehat{G} of cost $|\mathcal{S}| + |V(G)| + k$. Recall that $VC(G)$ is some optimal vertex cover in G and $q^*(\mathcal{S})$ is the cost of the optimal secluded Steiner tree in \widehat{G} . We now show that

$$q^*(\mathcal{S}) = |\mathcal{S}| + |V(G)| + |VC(G)| .$$

We begin by showing that $q^*(\mathcal{S}) \leq |\mathcal{S}| + |V(G)| + |VC(G)|$. Given a vertex cover C' for G , we show that there exists a Steiner tree $T \subseteq \widehat{G}$ of cost $\text{Cost}(T) \leq |\mathcal{S}| + |V(G)| + |C'|$. The tree T is given by $V(T) = \mathcal{S} \cup C'$ and $E(T) = E(\widehat{B}) \cup \{(\ell_i, v_i) \mid v_i \in C'\} \cup \{(v'_e, v_i), e = (v_i, v_j) \in E(G), v_i \in VC(G)\}$. Since C' is a legal vertex cover in G , it follows that every

terminal v'_e has at least one neighbor v_i (corresponding to the endpoints of e in G) that connects it to the tree T . Hence, T is a valid Steiner tree. We now analyze its cost. The cost of T consists of the set $V_1 = \mathcal{S} \cup V(G)$ of the terminals and their neighbors, and the set $V_2 = \{\widehat{v}_i \mid v_i \in C'\}$. Therefore, $\mathbf{Cost}(T) = |V_1| + |V_2| = |\mathcal{S}| + |V(G)| + |C'|$ and thus $q^*(\mathcal{S}) \leq |\mathcal{S}| + |V(G)| + |VC(G)|$ as required.

It remains to show that $|VC(G)| \leq q^*(\mathcal{S}) - (|\mathcal{S}| + |V(G)|)$. Given a Steiner tree T in \widehat{G} , we show how to obtain a legal vertex cover C such that $|C| \leq \mathbf{Cost}(T) - (|\mathcal{S}| + |V(G)|)$. Let $C = V(T) \cap V(G)$. First, note that C is a vertex cover. Assume for contradiction that it is not, and let $e = (v_i, v_j)$ be a non-covered edge by C . Since neither v_i nor v_j in C , we get that v'_e is disconnected in T , in contradiction to the fact that T is a Steiner tree. Finally, observe that $\mathbf{Cost}(T) = |\mathcal{S}| + |V(G)| + |V(G) \cap V(T)|$. Since $C = |V(G) \cap V(T)|$, part (a) of the theorem is established. We now turn to prove Thm. 3.4(b) and show that there exists a polynomial-time algorithm if the number of terminals is bounded by a constant.

3.2 Polynomial-time algorithm for Secluded Steiner Tree with $|\mathcal{S}| = O(1)$.

For simplicity we present the algorithm for maximum degree three (and note that the proof works for any fixed bounded degree). Let $k = |\mathcal{S}|$. We note that the main difference with **Secluded Steiner Tree** problem is that here we also need to take into account the “skeleton” of the tree. We iterate over possible skeletons of trees, where by skeleton of a tree we mean the following. Consider some tree T and contract every path $P \in \Sigma(T)$ such that all nodes in P but the endpoints are of degree 2 and are not terminal into a single edge. The resulting graph is the skeleton of the tree T . Note that it is enough to look on trees T such that all leaves are terminals (as otherwise we can take a subtree of the tree spanning all terminals). Note that the skeleton may contain at most k nodes that are not terminals. Given the (at most k) nodes that are not terminals, an upper bound on the number of different skeletons is the number of different trees with at most $2k$ nodes, which is $O((2k)^{2k-2})$. There are $O(n^k)$ options to choose the nonterminal nodes in the skeleton. We thus get that there are at most $O(n^k(2k)^{2k-2})$ different skeletons.

A subtree S_+ is an extended skeleton of the skeleton S if it obtained by replacing each edge $e_i = (x_i, y_i)$ in the skeleton S with a path P_i of the following form. Either the path P_i is a path of G of length at most 9 (and contains at most 8 internal nodes), or the prefix of the path P_i is a subpath of length 4 in G from x_i to some node z_i and the suffix of P_i

is a subpath of length 4 in G from some node w_i to y_i , the nodes z_i and w_i are connected by an imaginary edge (which will be replaced later by a path from G).

The algorithm for finding the tree with the lowest cost is as follows. Iterate over all possible skeletons S and for each skeleton S iterate over all possible extended skeleton S_+ . For each S_+ do the following. Let $V(S_+)$ be the set of nodes in S_+ . For each imaginary edge (x, y) in S_+ , consider the graph $G'(x, y)$ that is obtained by removing all nodes $V(S_+) \setminus \{x, y\}$ from G . Find the lowest cost **Secluded Path** $\hat{P}(x, y)$ from x to y in $G'(x, y)$. For each imaginary edge (x, y) in S_+ , replace the edge with the path $\hat{P}(x, y)$. Let $T(S_+)$ be the tree obtained through this process. Finally, return the tree $T(\hat{S}_+)$ with the lowest cost. Let T_{alg} be the returned tree.

We call nodes in a tree T splitting nodes if their degree in T is greater than 2. We call a node special in T if it is either a splitting node or a terminal. Consider the optimal tree T^* and let S^* be its skeleton and S_+^* be its extended skeleton. We claim that $\text{Cost}(T(S_+^*)) = \text{Cost}(T^*)$, this will imply the correctness of the algorithm.

As in the **Secluded Path** problem we may restrict ourselves only to trees such that all nodes that their parent are not special do not have edges to nodes on the tree that are not in their subtree, similarly, every node that none of whose children is special has no edges to nodes in its subtree (otherwise we can find a different tree with a subset of the nodes of T^*). In other words, all nodes in T^* that are neither special nor have a neighbor in T that is special have no edges in G to other nodes in T^* that are not adjacent to them in T^* .

The following observation is an extension of Obs. 3.5 and is crucial for the algorithm.

Observation 3.14 *Consider an edge e_1 in the skeleton S^* and their corresponding path $P_1 = (z_1, \dots, z_{r_1}) \in \Sigma(T^*)$. The node z_i for $6 \leq i \leq r_1 - 6$ cannot have a common neighbor with any other node $z \in T^* \setminus P_1$.*

Lemma 3.15 $\text{Cost}(T(S_+^*)) = \text{Cost}(T^*)$

Proof: The proof follows from Obs. 3.14. Note that in the optimal solution T^* , $N[S_+^*] \subseteq N[T^*]$. Consider an imaginary edge (x, y) in S_+^* . Let $P^*(x, y)$ be the subpath from x to y in T^* . Note that the subpath $P^*(x', y')$ are disjoint for all imaginary edges (x', y') in S_+^* . Let $c^*(x', y') = |N[P^*(x', y')] \setminus N[S_+^*]|$. Note that $\text{Cost}(T^*) = \sum_{(x', y')} c^*(x', y') + |N[S_+^*]|$. Note also that $|N[\hat{P}(x, y), G'(x, y)]| \leq c^*(x', y')$. We get that $\text{Cost}(T(S_+^*)) \leq \sum_{(x, y)} |N[\hat{P}(x, y), G'(x, y)]| + |N[S_+^*]| \leq \text{Cost}(T^*)$. ■

4 Secluded Connectivity for Specific Graph Families

Bounded-treewidth graphs. For a graph $G(V, E)$, let $TW(G)$ denote the *treewidth* of G . In this section, we consider graphs G of constant treewidth, i.e. $TW(G) = O(1)$. A *separation* of a graph $G = (V, E)$ is a triplet (A, B, S) where $A \cup B \cup S = V$, $A \cap B \subset S$ and $(u, v) \notin E$ for every $u \in A \setminus S$ and $v \in B \setminus S$. The set S then separates $A \setminus S$ and $B \setminus S$ in G . The concept of treewidth was introduced by Robertson and Seymour [22] using tree-decompositions. (See [2, 3] for an in-depth introduction to this topic.) For a graph $G = (V, E)$, a tree decomposition is a pair $(\mathcal{T}, \mathfrak{X})$ consisting of a tree $\mathcal{T} = (I, A)$ and a collection $\mathfrak{X} = \{X_i\}_{i \in I}$ of vertex subsets (called *bags*) with the following properties: (T1) Every vertex $v \in V(G)$ is contained in the least one bag X_i . For every edge $(u, v) \in E$ there is at least one bag X_i containing both vertices u, v . (T2) For every vertex $v \in V(G)$, the nodes i of \mathcal{T} with $v \in X_i$ form a subtree of \mathcal{T} .

To avoid confusion, the elements of I are referred to as *nodes* and the elements of $V(G)$ are referred to as *vertices*. In addition, we may informally refer to the node X_i by its index i . Let X_i^+ be the set containing the vertices of node i itself, and the vertices in the bags of its descendants in \mathcal{T} . Let U_i (respectively, U_i^+) be the set of terminals in X_i (respectively, X_i^+). Let ω be some upper bound on $TW(G)$ such that $|X_i| \leq \omega - 1$ for every $i \in I$. Let U_i^+ be the set of terminals in X_i^+ . We show the following.

Theorem 4.1 *The Secluded Steiner Tree problem (and hence also the Secluded Path problem) can be solved in linear time for graphs with fixed treewidth. In addition, given the tree decomposition of G , the Secluded Steiner Tree problem is solvable in $\tilde{O}(n^3)$ if $TW(G) = O(\log n / \log \log n)$. This holds even for weighted and directed graphs.*

If the tree decomposition is not known, then for fixed treewidth graphs it can be computed in linear time [1]. For simplicity, we consider the unweighted undirected case. The weighted and directed cases are considered by the end of the analysis. Given a tree T in G and a separation (A, B, S) in G , let $F_A = T(A)$ (respectively, $F_B = T(B)$) be the forest obtained by considering the edges of T restricted to A (respectively, B) nodes. Let $\text{Diff}(A, S) = N[T] \cap (A \setminus S)$ and $\text{Diff}(B, S) = N[T] \cap (B \setminus S)$. Finally, let $\text{Cost}(T, S) = N[T] \cap S$. Since (A, B, S) is a separation, it follows that

$$\text{Cost}(T) = \text{Diff}(A, S) + \text{Diff}(B, S) + \text{Cost}(T, S). \quad (32)$$

Such decomposition of the cost function $\text{Cost}(G')$ holds for any connectivity construct $G' \subseteq G$. Indeed, the correctness of the dynamic programming applied on the tree-

decomposition of G is established due to this cost independency property of Eq. (32).

4.0.1 Algorithm

The **Secluded Steiner Tree** algorithm is an extension of the algorithm of [6] for computing an optimal **Steiner Tree** in a bounded-treewidth graph. The input to the algorithm is a connected graph $G = (V, E)$, a set of terminals \mathcal{S} and a tree decomposition $\mathcal{T} = (I, A)$ of G . For ease of analysis, we assume that the tree decomposition \mathcal{T} is *nice* in the sense that every node $i \in I$ has at most two children $i_1, i_2 \in I$ in the tree decomposition \mathcal{T} (as in [6]).

Initially choose any root node $r \in I$ that contains at least one terminal, i.e. $X_r \cap \mathcal{S} \neq \emptyset$, and direct the tree \mathcal{T} such that r is the root. For each $i \in I$ we maintain a table $tb(i)$. Each entry $\kappa_i \in tb(i)$ in the table represents a configuration, implicitly corresponding to a subsolution. Such a subsolution is a forest $F \subseteq G$ that when imposed on the nodes of X_i^+ , resulting in $F(X_i^+)$, spans the terminals U_i^+ in X_i^+ , and $F(X_i^+)$ may be extended into a tree that spans all the terminals. Specifically, the configurations define a collection of disjoint subsets of the vertices of X_i and a collection of edges between vertices in X_i . In addition, it specifies the vertices $\tilde{v} \in X_i$ that are neighbors of the forest F (note that for such \tilde{v} , its neighbor in F is not necessarily in X_i ; it might be in an ancestor of X_i in \mathcal{T}). We follow the representation scheme of [6] where each subsolution is represented by a color assignment κ_i to the vertices of X_i , and extend it by adding an additional color ω that represents the status of being a neighbor. We refer to this coloring as a *configuration*. The interpretation of the colors is representing connected components in $F(X_i^+)$ (the forest F restricted to the vertices of X_i^+) and vertices in X_i that have neighbors in F , i.e., vertices $u \in X_i$ with the same color $\kappa_i(u) = c \in [1, \omega - 1]$ belong to the same connected component in the subsolution forest $F(X_i^+)$. The vertices u such that $\kappa_i(u) = \omega$ do not participate in the forest but have at least one neighbor in F . Finally, vertices u with $\kappa_i(u) = 0$ neither belong to the subsolution represented by κ_i nor have a neighbor that belongs to it. The idea behind this representation is the following. We assume that there is a Steiner tree T in G . The forest $T(X_i^+)$ is obtained by restricting the 1-neighborhood of the tree T to the vertices in X_i^+ . This restriction of $N[T] \cap X_i^+$ is encoded by the configuration κ_i . In particular, for a Steiner tree T , there is a unique configuration κ_i for each bag X_i that is consistent with T (as formally defined later). Overall, each table X_i has at most $B_{k'}$ entries, for $k' = \omega + 1$, where $B'_k = O(k')$ is the *Bell-number* [6], corresponding to the number of partitions of at most ω elements into $\omega + 1$ classes.

Legal Configuration. Let $A(\kappa_i) = \{u \in X_i \mid \kappa_i(u) > 0\}$ denote the set of active vertices in the configuration κ_i . A configuration κ_i is *legal* if it satisfies (L1) every terminal $u \in U_i$ is assigned a color in the range $\kappa_i(u) \in [1, \omega]$, and (L2) $N[A(\kappa_i)] \cap X_i \subseteq A(\kappa_i)$.

The algorithm proceeds by computing, for each legal configuration $\kappa_i \in tb(i)$, a value $Val(\kappa_i)$ that represents the minimum number of vertices in X_i^+ that belongs to the neighborhood of a forest F that respects the configuration. These values are computed bottom-up. Before we proceed, we first define the notion of compatible configurations.

Compatible configurations. Let i be parent of j in the tree \mathcal{T} . Then $\kappa_i \in tb(i)$ is *compatible* with $\kappa_j \in tb(j)$, denoted as $\kappa_i \sim \kappa_j$ if the following four properties hold:

- (Q1) κ_i and κ_j agree on their mutual vertices $X' = X_i \cap X_j$. For any $u \in X'$, $\kappa_i(u) = \omega$ iff $\kappa_j(u) = \omega$. In addition, for any pair $u, v \in X'$ that are in the same connected component, $\kappa_j(u) = \kappa_j(v) \in [1, \omega - 1]$, it holds that u and v are connected in κ_i as well. (Note that some components in the child j might be connected in the parent i , and thus the converse does not hold.) (Q2) If κ_i connects two connected components of κ_j , then this is supported by the graph G edges (i.e., the the edges required for the connectivity exist). (Q3) Any terminal of X_j is connected to some connected component in the forest represented by κ_i . Hence, κ_i contains at most $|X_i|$ connected components and for each terminal $u \in X_j$ there exists some $v \in X_i \cap X_j$ such that $\kappa_j(v) = \kappa_j(u)$ and $\kappa_i(u) \in [1, \omega - 1]$. (Q4) $N[A(\kappa_j)] \cap X_i \subseteq A(\kappa_i)$.

The value $Val(\kappa_i)$ of the coloring $\kappa_i \in tb(i)$ is computed bottom-up. Let X_i be a leaf. Then for every *legal* $\kappa_i \in tb(i)$, set $Val(\kappa_i) = |A(\kappa_i)|$ and $Val(\kappa_i) = \infty$ if the configuration is invalid. Consider a non-leaf node X_i with children X_{i_1} and X_{i_2} . Let $\Psi = (\kappa_i, \kappa_{i_1}, \kappa_{i_2})$ be a triple of compatible configurations, such that $\kappa_{i_1} \in tb(i_1)$ and $\kappa_{i_2} \in tb(i_2)$. Define

$$Val(\Psi) = Val(\kappa_{i_1}, \kappa_i) + Val(\kappa_{i_2}, \kappa_i) + |A(\kappa_i)|, \quad (33)$$

where, for $j \in \{i_1, i_2\}$

$$Val(\kappa_j, \kappa_i) = Val(\kappa_j) - |A(\kappa_i) \cap A(\kappa_j)|. \quad (34)$$

Since $(X_{i_1}^+, X_{i_2}^+, X_i)$ is a separation in the graph $G(X_i^+)$, the correctness of Eq. (33) follows from Eq. (32). Finally, for each entry $\kappa_i \in tb(i)$, maintain the minimum cost over any set of compatible triples. Define,

$$Val(\kappa_i) = \min_{\Psi = (\kappa_i, \kappa_{i_1}, \kappa_{i_2})} \{Val(\Psi) \mid \kappa_{i_1} \in tb(i_1), \kappa_{i_2} \in tb(i_2), \kappa_{i_1} \sim \kappa_i, \kappa_{i_2} \sim \kappa_i\}. \quad (35)$$

The optimal solution value for the whole graph can be found in the root bag X_r of \mathcal{T} , identifying a cheapest solution where all vertices with color $\neq 0$ are contained in the same

connected component (i.e., have the same color). Computing the optimal solution (which is sufficiently represented by the set of nodes in the final tree) is possible by backtracking or by storing the set of edges for each row and each bag.

4.0.2 Analysis

For a given configuration entry $\kappa_i \in tb(i)$, the forest F agrees with κ_i if the following four properties hold:

(Z1) $F(X_i^+)$ has C connected components where $C = |\{\kappa(u) \in [1, \omega - 1], u \in X_i\}|$ is the number of distinct colors in the coloring κ_i .

(Z2) All vertices $u \in X_i$ of the same color, $\kappa_i(u)$ are in the same connected component in $F(X_i^+)$.

(Z3) The X_i neighbors of the forest are active, $N[F] \cap X_i = A(\kappa_i)$.

(Z4) The terminals U_i^+ are connected in $F(X_i^+)$ in the following manner. Any terminal $u \in U_i^+ \setminus X_i$ is in *non-singleton* component in F . Specifically, it is connected to some node $v \in X_i$. In addition, the terminals of $U_i^+ \cap X_i$, are in some connected component in F .

For a legal configuration κ_i define the family of all forests in G that agree with κ_i as

$$\mathcal{F}(\kappa_i) = \{F \subseteq G \mid F \text{ agree with } \kappa_i\} \quad (36)$$

Let

$$Val(F, X_i) = |N[F] \cap X_i^+| \quad (37)$$

and $Val^*(\kappa_i) = \min\{Val(F, X_i) \mid F \in \mathcal{F}(\kappa_i)\}$ be the minimum value of any forest $F \in \mathcal{F}(\kappa_i)$.

Lemma 4.2 $Val(\kappa_i) = Val^*(\kappa_i)$.

Proof: This can be shown by a straightforward inductive proof on the decomposition tree. The base cases are leaf nodes where the hypothesis clearly holds. Consider $i \geq 1$ and assume the induction assumption holds for all descendants of bag X_i . Let $F^* \in \mathcal{F}(\kappa_i)$ be the forest that attains the minimum value in the family $\mathcal{F}(\kappa_i)$, i.e., $Val(F^*, X_i) = Val^*(\kappa_i)$. Let X_1, X_2 be the children of X_i in \mathcal{T} and let $\kappa_1 \in tb(1), \kappa_2 \in tb(2)$ be the two compatible entries such that $F^* \in \mathcal{F}(\kappa_1) \cap \mathcal{F}(\kappa_2)$, hence $N[F^*] \cap X_j = A(\kappa_j)$ for $j \in \{1, 2, i\}$. For a forest $F \in \mathcal{F}(\kappa_j)$ and $j \in \{1, 2, i\}$, define

$$Diff(F, X_j, X_i) = |N[F] \cap (X_j^+ \setminus X_i)|. \quad (38)$$

Since (X_1^+, X_2^+, X_i) is a separation in $G(X_i^+)$ and by Property (Q4) we have the following.

Claim 4.1 $\text{Diff}(F, X_j, X_i) = \text{Val}(F, X_j) - |A(\kappa_i) \cap A(\kappa_j)|$ for every $F \in \mathcal{F}(\kappa_j)$, where $j \in \{1, 2\}$.

In addition, since (X_1^+, X_2^+, X_i) is a separation in $G(X_i^+)$ by Eq. (32), it follows that

$$\text{Val}(F^*, X_i) = \text{Diff}(F^*, X_1, X_i) + \text{Diff}(F^*, X_2, X_i) + |A(\kappa_i)|. \quad (39)$$

For $j \in \{1, 2\}$, let $F_j^* \in \mathcal{F}(\kappa_j)$ be the forest that attains the minimum value of $\text{Val}(F_j^*, \kappa_j)$ in their family, i.e., $\text{Val}^*(\kappa_j) = \text{Val}(F_j^*, \kappa_j)$. By induction assumption, $\text{Val}(\kappa_j) = \text{Val}(F_j^*, \kappa_j)$. By Eq. (34) and Claim 4.1, we get that

$$\text{Val}(\kappa_j, \kappa_i) = \text{Val}(\kappa_j) - |A(\kappa_i) \cap A(\kappa_j)| \quad (40)$$

$$= \text{Val}(F_j^*, \kappa_j) - |A(\kappa_i) \cap A(\kappa_j)| \quad (41)$$

$$= \text{Diff}(F_j^*, X_j, X_i), \quad (42)$$

where Ineq. (40) follows from Eq. (34), Ineq. (41) follows from induction assumption and Ineq. (42) follows from Claim 4.1. Thus since the triple $\Psi = (\kappa_i, \kappa_1, \kappa_2)$ consists of compatible configurations, by Eq. (33), Eq. (34), Eq. (35) and Eq. (40) we get that

$$\text{Val}(\kappa_i) \leq \text{Val}(\Psi) = \text{Diff}(F_1^*, X_1, X_i) + \text{Diff}(F_2^*, X_2, X_i) + |A(\kappa_i)|. \quad (43)$$

Assume for contradiction that $\text{Val}(\kappa_i) > \text{Val}^*(\kappa_i)$. Then, by Eq. (43) and Eq. (39), we get that there exists $j \in \{1, 2\}$ such that

$$\text{Diff}(F_j^*, X_j, X_i) > \text{Diff}(F^*, X_j, X_i). \quad (44)$$

Without loss of generality let $j = 1$. Combining Claim 4.1 and Ineq. (44), we get that $\text{Val}(F^*, X_1) < \text{Val}(F_1^*, X_1)$, and since $F^* \in \mathcal{F}(\kappa_1)$, we end with contradiction to the induction assumption. We therefore get that $\text{Val}(F^*, X_1) = \text{Val}(F_1^*, X_1)$ and $\text{Val}(F^*, X_2) = \text{Val}(F_2^*, X_2)$. The lemma follows. ■

We now complete the proof of Thm. 4.1

Proof: Let $\Upsilon = \{\kappa_r \in \text{tb}(r) \mid \kappa_r(u) \in \{0, 1, \omega\}, \forall u \in X_r\}$ be set of configurations in the root r of \mathcal{T} , in which all vertices are in a single connected component. Note that there exists a configuration κ_r such that the optimal secluded tree $T^* \in \mathcal{F}(\kappa_r)$. Let $\kappa_r^* \in \Upsilon$ be the configuration that attains the minimum value $\text{Val}(\kappa_r^*)$ in the set. It then follows,

by Lemma 4.2, $Val(\kappa_r^*) = Val^*(\kappa_r^*) = q^*(\mathcal{S})$. The Steiner tree can be computed by a brute-force backtracking procedure.

We now turn to running time. The size of any table $tb(i)$ is bounded by $O(\omega! \cdot \omega)$. During the bottom-up traversal of \mathcal{T} we consider all possible row combinations for at most 3 tables. The merge operation can be done in $O(\omega)$ time, and hence requires overall $O(B_{\omega+1}^3 \cdot \omega)$. All other operations for entry processing are linear in the size of the bag, $O(\omega)$. Note that computing the values of subsolution κ_i are linear in the treewidth. Thus our time complexity is of the same order as that of [6]. In particular, it is linear for graphs with fixed treewidth, in this case the tree decomposition can also be computed in linear time and hence the total complexity is linear. If $TW(G) = O(\log n / \log \log n)$ then given the tree decomposition \mathcal{T} for G , the running time is $\tilde{O}(n^3)$. ■

Note that the algorithm extends also to directed and node-weighted graphs. We now outline the modifications needed for these extensions. The extension to weighted graph is immediate. The only modification needed is to define $\text{Diff}(A, S)$, $\text{Diff}(B, S)$ and $\text{Cost}(T, S)$ according to the weighted cost as in Eq. (1) instead of the unweighted cost, and apply a similar analysis.

The extension to directed graphs is a bit more involved. Note that in the directed case it is not enough to store configurations that represent the different connected components since in order to construct a valid tree we need to make sure that every node has only one parent. Therefore it is important to “keep track” of the roots of these connected components. For each connected component C we store as part of the configuration the root $r(C)$ of this connected component. Namely, we add to each configuration a root for each color. In order to merge two connected components C_1 and C_2 to single tree with root $r(C_1)$, we only consider edges from $r(C_2)$ to C_1 . However, note that the root of a component does not necessarily have to be from the current bag and hence the number of different configurations might be too large. To overcome this issue we use the following observation. Consider a bag X_i and a configuration $\kappa_i \in tb(i)$. Note that it cannot be that two different connected components in κ_i would have both roots that are not from X_i . To see this, consider a subtree whose root r is not in X_i , since X_i is a separator then r does not have edges to nodes in $V \setminus X_i^+$ forcing r to be the root of the final tree. Therefore, there could be at most one such component. We thus restrict ourselves to configurations where there is at most one connected component with a root not in X_i (otherwise the configuration is invalid). Notice that the number of valid configurations might increase by at most a factor of n (for all options of the root not in X_i). Hence, the running time increases by a factor of n . The rest of the analysis is similar and thus is omitted from this

version.

Bounded Density Graphs. Let $\text{DegCost}^*(\mathcal{S}) = \min\{\text{DegCost}(T) \mid T \in \mathcal{T}(\mathcal{S})\}$. We show the following.

Proposition 4.3 *Let \mathcal{G} be a hereditary class of graphs with a linear number of edges, i.e., a set of graphs such that for each $G \in \mathcal{G}$, $|E(G)| \leq \ell \cdot |V(G)|$ for some constant ℓ , and where $G \in \mathcal{G}$ implies that $G' \in \mathcal{G}$ for each subgraph G' of G . Then $\text{DegCost}^*(\mathcal{S}) \leq 2\ell \cdot q^*(\mathcal{S})$.*

Proof: Note that for every Steiner tree T in G , it holds that

$$\text{Cost}(T) \leq \text{DegCost}(T) \leq 2\ell \cdot \text{Cost}(T). \quad (45)$$

The left inequality is immediate. We now prove the right inequality. Let $N = N[T]$ be the neighborhood of T and let $G(N)$ be the subgraph of G induced by N and $E(N)$ be its edge set. We have that

$$\text{DegCost}(T) = 2|E(N)| \leq 2\ell|N| = 2\ell \cdot \text{Cost}(T),$$

where the first equality follows by definition, the second inequality follows from the hereditary nature of \mathcal{G} and the third equality follows by the definition of $\text{Cost}(T)$. Hence, Eq. 45 holds. In particular, for the optimal secluded tree T^* , we get that

$$\text{Cost}(T^*) = q^*(\mathcal{S}) \leq 2\ell \cdot \text{DegCost}(T^*) \leq 2\ell \cdot \text{DegCost}^*(\mathcal{S}).$$

The proposition holds. **■**

An example of such a class of graphs is the family of planar graphs. For this family, the above proposition yields a 6-approximation for the **Secluded Path** problem. (A more careful direct analysis for planar graphs yields a 3-approximation, however) Finally, note that whereas computing $\text{DegCost}^*(\mathcal{S})$ for a constant number of terminals k is polynomial, for arbitrary k , computing $\text{DegCost}^*(\mathcal{S})$ is NP-hard but can be approximated to within a ratio of $\Theta(\log k)$ thanks to Claim 2.3. Thus, for the class of bounded density graphs, by Prop. 4.3, the **Secluded Path** problem has a constant ratio approximation and the **Secluded Steiner Tree** problem has an $\Theta(\log k)$ ratio approximation.

Theorem 4.4 *For the class of bounded-density graphs, the **Secluded Path** (respectively, **Secluded Steiner Tree**) problem is approximated within a ratio of $O(1)$ (resp., $\Theta(\log k)$).*

References

- [1] H.L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. In *STOC*, 226–234, 1993.
- [2] H.L. Bodlaender. A tourist guide through treewidth. *Acta Cybern.*, 11:1–22, 1993.
- [3] H.L. Bodlaender. Treewidth: Structure and algorithms. In *SIROCCO*, 11–25, 2007.
- [4] R.D. Carr, S. Doddi, G. Konjevod, and M.V. Marathe. On the red-blue set cover problem. In *SODA*, 345–353, 2000.
- [5] A. Chen, S. Kumar, and T.-H. Lai. Local barrier coverage in wireless sensor networks. *IEEE Tr. Mob. Comput.*, 9:491–504, 2010.
- [6] M. Chimani, P. Mutzel, and B. Zey. Improved steiner tree algorithms for bounded treewidth. In *IWOCA*, 374–386, 2011.
- [7] I. Dinur and S. Safra. On the hardness of approximating label-cover. *IPL*, 89:247–254, 2004.
- [8] U. Feige. A threshold of $\ln n$ for approximating set cover. *J. ACM*, 45:634–652, 1998.
- [9] M.R. Fellows, J. Guo, and I.A. Kanj. The parameterized complexity of some minimum label problems. *JCSS*, 76:727–740, 2010.
- [10] M.R. Garey and D.S. Johnson. The Rectilinear Steiner Tree Problem is NP-Complete. *SIAM J. Appl. Math.*, 32:826–834, 1977.
- [11] R. Hassin, J. Monnot, and D. Segev. Approximation algorithms and hardness results for labeled connectivity problems. *J. Comb. Optim.*, 14:437–453, 2007.
- [12] A. Johansson and P. Dell’Acqua. Knowledge-based probability maps for covert pathfinding. In *MIG*, 339–350, 2010.
- [13] R.M. Karp. Reducibility among combinatorial problems. In R.E. Miller and J.W. Thatcher, Eds, *Complexity of Computer Computations*, 85–103. Plenum Press, NY, 1972.
- [14] P.N. Klein and R. Ravi. A nearly best-possible approximation algorithm for node-weighted steiner trees. *J. Algo.*, 19:104–115, 1995.
- [15] S.O. Krumke and H.-C. Wirth. On the minimum label spanning tree problem. *IPL*, 66:81–85, 1998.
- [16] B. Liu, O. Dousse, J. Wang, and A. Saipulla. Strong barrier coverage of wireless sensor networks. In *MobiHoc*, 411–420, 2008.

- [17] M. Marzouqi and R. Jarvis. New visibility-based path-planning approach for covert robotic navigation. *Robotica*, 24:759–773, 2006.
- [18] M. Marzouqi and R. Jarvis. Robotic covert path planning: A survey. In *RAM*, 77–82, 2011.
- [19] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In *INFOCOM*, 1380–1387, 2001.
- [20] J. Monnot. The labeled perfect matching in bipartite graphs. *IPL*, 96:81–88, 2005.
- [21] D. Peleg. Approximation algorithms for the label-cover_{max} and red-blue set cover problems. *J. Discrete Algo.*, 5:55–64, 2007.
- [22] N. Robertson and P.D. Seymour. Graph minors. ii. algorithmic aspects of tree-width. *J. Algo.*, 7:309–322, 1986.
- [23] S. Yuan, S. Varma, and J.P. Jue. Minimum-color path problems for reliability in mesh networks. In *INFOCOM*, 2658–2669, 2005.
- [24] P. Zhang, J.Y. Cai, L. Tang, and W. Zhao. Approximation and hardness results for label cut and related problems. *J. Comb. Optim.*, 21:192–208, 2011.