Parameterized Complexity of Secluded Connectivity Problems[*]

Fedor V. Fomin[1,2], Petr A. Golovach[1,2], Nikolay Karpov[2], and Alexander S. Kulikov[2]

1   Department of Informatics, University of Bergen, Norway,
     {fedor.fomin,petr.golovach}@uib.no
2   Steklov Institute of Mathematics at St.Petersburg, Russian Academy of
     Sciences, Russia, {kimaska,alexander.s.kulikov}@gmail.com

Abstract

The SECLUDED PATH problem introduced by Chechik et al. in [ESA 2013] models a situation where a sensitive information has to be transmitted between a pair of nodes along a path in a network. The measure of the quality of a selected path is its *exposure*, which is the total weight of vertices in its closed neighborhood. In order to minimize the risk of intercepting the information, we are interested in selecting a *secluded* path, i.e. a path with a small exposure. Similarly, the SECLUDED STEINER TREE problem is to find a tree in a graph connecting a given set of terminals such that the exposure of the tree is minimized. In this work, we obtain the following results about parameterized complexity of secluded connectivity problems.

We start from an observation that being parameterized by the size of the exposure, the problem is fixed-parameter tractable (FPT). More precisely, we give an algorithm deciding if a graph G with a given cost function $\omega \colon V(G) \to \mathbb{N}$ contains a secluded path of exposure at most k with the cost at most C in time $\mathcal{O}(3^{k/3} \cdot (n+m) \log W)$, where W is the maximum value of ω on an input graph G. Similarly, SECLUDED STEINER TREE is solvable in time $\mathcal{O}(2^k k^2 \cdot (n+m) \log W)$.

The main result of this paper is about "above guarantee" parameterizations for secluded problems. We show that SECLUDED STEINER TREE is FPT being parameterized by $r + p$, where p is the number of the terminals, ℓ the size of an optimum Steiner tree, and $r = k - \ell$. We complement this result by showing that the problem is co-W[1]-hard when parameterized by r only.

We also investigate SECLUDED STEINER TREE from kernelization perspective and provide several lower and upper bounds when parameters are the treewidth, the size of a vertex cover, maximum vertex degree and the solution size. Finally, we refine the algorithmic result of Chechik et al. by improving the exponential dependence from the treewidth of the input graph.

1998 ACM Subject Classification G.2.2 Graph Theory, F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases Secluded path, Secluded Steiner tree, parameterized complexity

1   Introduction

SECLUDED PATH and SECLUDED STEINER TREE problems were introduced in Chechik et al. in [8]. In the SECLUDED PATH problem, for given vertices s and t of a graph G, the task is to

---

find an s, t-path with the minimum *exposure*, i.e. a path P such that the number of vertices from P plus the number of vertices of G adjacent to vertices of P is minimized. The name secluded comes from the setting where one wants to transfer a confident information over a path in a network which can be intercepted either while passing through a vertex of the path or from some adjacent vertex. Thus the problem is to select a secluded path minimizing the risk of interception of the information. When instead of connecting two vertices one needs to connect a set of terminals, we arrive naturally to the SECLUDED STEINER TREE.

More precisely, SECLUDED STEINER TREE is the following problem.

SECLUDED STEINER TREE
Input: A graph G with a cost function $\omega \colon V(G) \to \mathbb{N}$, a set $S = \{s_1, \ldots, s_p\} \subseteq V(G)$ of terminals, and non-negative integers k and C.
Question: Is there a connected subgraph T of G with $S \subseteq V(T)$ such that $|N_G[V(T)]| \leq k$ and $\omega(N_G[V(T)]) \leq C$?

If $\omega(v) = 1$ for each $v \in V(G)$ and $C = k$, then we have an instance of SECLUDED STEINER TREE without costs; respectively, we omit ω and C whenever we consider such instances.

Clearly, it can be assumed that T is a tree, and thus the problem can be seen as a variant of the classical STEINER TREE problem. For the special case $p = 2$, we call the problem SECLUDED PATH.

Previous work. The study of the secluded connectivity was initiated by Chechik et al. [7, 8] who showed that the decision version of SECLUDED PATH without costs is NP-complete. Moreover, for the optimization version of the problem, it is hard to approximate within a factor of $\mathcal{O}(2^{\log^{1-\varepsilon} n})$, n is the number of vertices in the input graph, for any $\varepsilon > 0$ (under an appropriate complexity assumption) [8]. Chechik et al. [8] also provided several approximation and parameterized algorithms for SECLUDED PATH and SECLUDED STEINER TREE. Interestingly, when there are no costs, SECLUDED PATH is solvable in time $\Delta^\Delta \cdot n^{\mathcal{O}(1)}$, where Δ is the maximum vertex degree and and thus is FPT being parameterized by Δ. Chechik et al. [8] also showed that when the treewidth of the input graph does not exceed t, then the SECLUDED STEINER TREE problem is solvable in time $2^{\mathcal{O}(t \log t)} \cdot n^{O(1)} \cdot \log W$,[1] where W is the maximum value of ω on an input graph G. Johnson et al. [18] obtained several approximation results for SECLUDED PATH and showed that the problem with costs is NP-hard for subcubic graphs improving the previous result of Chechik et al. [8] for graphs of maximum degree 4.

The problems related to secluded path and connectivity under different names were considered by several authors. Motivated by secure communications in wireless ad hoc networks, Gao et al.[15] introduced the very similar notion of the thinnest path. The motivation of Gilbers [17], who introduced the problem under the name of the minimum witness path, came from the study of art gallery problems.

Our results. In this paper we initiate the systematic study of both problems from the Parameterized Complexity perspective and obtain the following results. In Section 3, we start from observations that SECLUDED PATH and SECLUDED STEINER TREE are FPT when parameterized by the size of the solution k by giving algorithms of running time

---

[1] In fact, Chechik et al. [8] give the algorithm that finds a tree with the exposure of minimum cost, but the algorithm can be easily modified for the more general SECLUDED STEINER TREE.

$\mathcal{O}(3^{k/3} \cdot (n+m) \log W)$ and $\mathcal{O}(2^k k^2 \cdot (n+m) \log W)$, where W is the maximum value of ω on an input graph G, correspondingly.

We consider the "above guarantee" parameterizations of both problems in Section 4. Recall that if $s_1, \ldots, s_p$ are vertices of a graph G, then a connected subgraph T of G of minimum size such that $s_1, \ldots, s_p \in V(T)$ is called a *Steiner tree* for the terminals $s_1, \ldots, s_p$. If $p = 2$, then a Steiner tree is a shortest (s_1, s_2)-path. Clearly, if ℓ is the size (the number of vertices) of a Steiner tree, then for any connected subgraph T of G with $S \subseteq V(T)$, $|N_G[V(T)]| \geq \ell$. Recall that the STEINER TREE problem is well known to be NP-complete as it is included in the famous Karp's list of 21 NP-complete problems [19], but in 1971 Dreyfus and Wagner [12] proved that the problem can be solved in time $O^*(3^p)$, i.e., it is FPT when parameterized by the number of terminals. The currently best FPT-algorithms for STEINER TREE running in time $O^*(2^p)$ are given by Björklund et al. [2] and Nederlof [21] (the first algorithm demands exponential in p space and the latter uses polynomial space). In Section 4 we show that SECLUDED PATH and SECLUDED STEINER TREE are FPT when the problems are parameterized by $r + p$, where $r = k - \ell$. From the other side, we show that the problem is co-W[1]-hard when parameterized by r only.

In Section 5, we provide a thorough study of the kernelization of the problem from the structural paramaterization perspective. We consider parameterizations by the treewidth, size of the solution, maximum degree and the size of a vertex cover of the input graph. We show that it is unlikely that SECLUDED PATH (even without costs) parameterized by the solution size, the treewidth and the maximum degree of the input graph, admits a polynomial kernel. In particular, this complements the FPT algorithmic findings of Chechik et al. [8] for graphs of bounded treewidth and of bounded maximum vertex degree. The same holds for the "above guarantee" parameterization instead the solution size as well. On the other hand, we show that SECLUDED STEINER TREE has a polynomial kernel when parameterized by k and the vertex cover number of the input graph. Interestingly, when we parameterize only by the vertex cover number, again, we show that most likely the problem does not admit a polynomial kernel. Finally, we refine the algorithm on graphs of bounded treewidth of Chechik et al. [8] by showing that SECLUDED STEINER TREE without costs can be solved by a randomized algorithm in time that single-exponentially depends on treewidth by applying the Count & Color technique of Cygan et al. [10] and further observe that for the general variant of the problem with costs, the same Count & Color technique can be used as well and also a single-exponential deterministic algorithm can be obtained by making use the representative set technique developed by Fomin et al. [14].

Due to space restrictions some proofs are omitted in this extended abstract. The full version of the paper is available in [13].

2   Basic definitions and preliminaries

We consider only finite undirected graphs without loops or multiple edges. The vertex set of a graph G is denoted by $V(G)$ and the edge set is denoted by $E(G)$. Throughout the paper we typically use n and m to denote the number of vertices and edges respectively.

For a set of vertices $U \subseteq V(G)$, $G[U]$ denotes the subgraph of G induced by U. For a vertex v, we denote by $N_G(v)$ its *(open) neighborhood*, that is, the set of vertices which are adjacent to v, and for a set $U \subseteq V(G)$, $N_G(U) = (\cup_{v \in U} N_G(v)) \setminus U$. The *closed neighborhood* $N_G[v] = N_G(v) \cup \{v\}$. Respectively, $N_G[U] = N_G(U) \cup U$. For a set $U \subseteq V(G)$, $G - U$ denotes the subgraph of G induced by $V(G) \setminus U$. If $U = \{u\}$, we write $G - u$ instead of $G - \{u\}$. The *degree* of a vertex v is denoted by $d_G(v) = |N_G(v)|$. We say that a vertex

v is *pendant* if $d_G(v) = 1$. A vertex v of a connected graph G with at least 2 vertices is a *cut vertex* if $G - u$ is disconnected. A connected graph G is *biconnected* if it has at least 2 vertices and has no cut vertices. A *block* of a connected graph G is an inclusion-maximal biconnected subgraph of G. A block is *trivial* if it has exactly 2 vertices. We say that vertex set X is connected if $G[X]$ is connected.

Parameterized complexity is a two dimensional framework for studying the computational complexity of a problem. One dimension is the input size n and another one is a parameter k. It is said that a problem is *fixed parameter tractable* (or FPT), if it can be solved in time $f(k) \cdot n^{O(1)}$ for some function f. A *kernelization* for a parameterized problem is a polynomial algorithm that maps each instance (x, k) with the input x and the parameter k to an instance (x', k') such that i) (x, k) is a yes-instance if and only if (x', k') is a yes-instance of the problem, and ii) the size of x' is bounded by $f(k)$ for a computable function f. The output (x', k') is called a *kernel*. The function f is said to be a *size* of a kernel. Respectively, a kernel is *polynomial* if f is polynomial. While a parameterized problem is FPT if and only if it has a kernel, it is widely believed that not all FPT problems have polynomial kernels. In particular, Bodlaender et al. [4, 5] introduced techniques that allow to show that a parameterized problem has no polynomial kernel unless NP $\subseteq$ coNP /poly. We refer to the recent books of Cygan et al. [9] and Downey and Fellows [11] for detailed introductions to parameterized complexity.

We use randomized algorithms for our problems. Recall that a *Monte Carlo algorithm* is a randomized algorithm whose running time is deterministic, but whose output may be incorrect with a certain (typically small) probability. A Monte-Carlo algorithm is *true-biased* (*false-biased* respectively) if it always returns a correct answer when it returns a yes-answer (a no-answer respectively).

3 FPT-algorithms for the problems parameterized by the solution size

In this section we consider SECLUDED PATH and SECLUDED STEINER TREE problems parameterized by the size of the solution, i.e., by k. We also show how these parameterized algorithms can be used to design faster exact exponential algorithms.

We start with SECLUDED PATH.[2]

▶ **Theorem 1** (∗)**.** SECLUDED PATH *is solvable in time* $\mathcal{O}(3^{k/3} \cdot n \log W)$, *where* W *is the maximum value of* ω *on an input graph* G.

For SECLUDED STEINER TREE we prove the following theorem.

▶ **Theorem 2** (∗)**.** SECLUDED STEINER TREE *can be solved in time* $\mathcal{O}(2^k k^2 \cdot (n + m) \log W)$, *where* W *is the maximum value of* ω *on an input graph* G.

Parameterized algorithms for SECLUDED PATH and SECLUDED STEINER TREE combined with a brute-force procedure imply the following exact exponential algorithms for the problems.

▶ **Theorem 3** (∗)**.** *On an* n*-vertex graph,* SECLUDED PATH *is solvable in time* $\mathcal{O}(1.3896^n \cdot \log W)$ *and* SECLUDED STEINER TREE *is solvable in time* $\mathcal{O}(1.7088^n \cdot \log W)$, *where* W *is the maximum value of* ω *on an input graph* G.

---

[2] The proofs of statements marked with (∗) can be found in [13].

FPT-algorithms for the problems parameterized above the guaranteed value

In this section we show that SECLUDED PATH and SECLUDED STEINER TREE are FPT when the problems are parameterized by $r + p$ where $r = k - \ell$ and ℓ is the size of a Steiner tree for S.

▶ **Theorem 4** (∗). SECLUDED PATH *is solvable in time $\mathcal{O}(2^{k-\ell} \cdot (n+m) \log W)$, where ℓ is the length of a shortest (u,v)-path for $\{u,v\} = S$ and W is the maximum value of ω on an input graph G.*

We need some structural properties of solutions of SECLUDED STEINER TREE. We start with an auxiliary lemma bounding the number of vertices of degree at least three in the subgraph of G induced by a solution as well as the number of their neighbors in this subgraphs.

▶ **Lemma 5.** *Let G be a connected graph and $S \subseteq V(G)$, $p = |S|$. Let F be an inclusion minimal connected induced subgraph of G such that $S \subseteq V(F)$ and $X = \{v \in V(F) | d_F(v) \geq 3\} \cup S$. Then (i) $|X| \leq 4p - 6$, and (ii) $|N_F(X)| \leq 4p - 6$.*

Proof. Let $\mathcal{B}$ be the set of blocks of F. Consider a bipartite graph T with the bipartition $(V(F), \mathcal{B})$ of the vertex such that $v \in V(F)$ and $b \in \mathcal{B}$ are adjacent if and only if v is a vertex of b. Notice that T is a tree. Recall that the *vertex dissolution* operation for a vertex v of degree 2 deletes v together with incident edges and replaces them by the edge joining the neighbors of v. Denote by T' the tree obtained from T by consequent dissolving all vertices of T of degree 2 that are not in S. Denote by L the set of leaves of T. By the minimality of F, $L \subseteq S$. Let $q_1 = |L| \leq p$, and let q_2 be the number of degree 2 vertices and q_3 be the number of vertices of degree at least 3 in T. Clearly, $q_1 + 2q_2 + 3q_3 \leq 2|E(T)| = 2(q_1 + q_2 + q_3 - 1)$. Then $q_3 \leq q_1 - 2 \leq p - 2$. We have that $|\{v \in V(T) | d_T(v) \geq 3\} \cup S| \leq q_3 + p \leq 2p - 2$ and $|V(T')| \leq 2p - 2$. Observe that if $d_F(v) \geq 3$ for $v \in V(F) \setminus S$, then v is a cut vertex of F and either v is included in at least 3 blocks of F, or v is in a block of size at least 3. In the second case, v is adjacent to a vertex $b \in \mathcal{B}$ of T with degree at least 3. It implies that $|X| \leq 2|E(T')| = 2(|V(T')| - 1) \leq 4p - 6$ and we have (i). To show (ii), observe that $|N_F(X)| \leq 2|E(T')| \leq 4p - 6$. ◀

The following lemma provides a bound on the number of vertices of a tree that have neighbors outside the tree.

▶ **Lemma 6.** *Let G be a connected graph and $S \subseteq V(G)$, $p = |S|$. Let ℓ be the size of a Steiner tree for S and r be a positive integer. Suppose that T is an inclusion minimal subgraph of G such that T is a tree spanning S and $|N_G[V(T)]| \leq \ell + r$. Then for $Y = N_G(V(T))$, $|N_G(Y) \cap V(T)| \leq 4p + 2r - 5$.*

Proof. Denote by L the set of leaves of T and by D the set of vertices of degree at least 3 in T. Clearly, $L \subseteq S$. We select a leaf z of T as the *root* of T. The selection of a root defines a parent-child relation on T. We order the vertices of T by the increase of their distances to z in T; the vertices on the same distance are ordered arbitrarily. Denote the obtained linear order by $\prec$. For each $u \in Y$, denote by $x(u)$ the unique minimum vertex in $N_G(u) \cap V(T)$ with respect to $\prec$. Let $U = \{x(u) | u \in Y\}$. For a vertex $u \in Y$ and $v \in N_G(u) \cap V(T) \setminus \{x(u)\}$, let $y(u,v)$ be the parent of v in T. Let $W = \{y(u,v) | u \in Y, v \in N_G(u) \cap V(T), v \neq x(u)\}$ and $W' = W \setminus (S \cup D \cup U)$.

Let $F = G[V(T) \cup Y]$.

▶ **Claim 7.** *Set $F' = F - W'$ is connected.*

Proof of the claim. Since all leaves of T including z are in S, we have that $z \in V(F')$. To prove the claim, we show that for each vertex $v \in V(F')$, there is a (v, z)-path in F'. Every vertex $u \in Y$ has a neighbor $x(u)$ in F'. Hence, it is sufficient to prove the existence of (v, z)-paths for $v \in V(T) \setminus W'$. The proof is by induction on the number of v with respect to $\prec$. The first vertex is z, and if $v = z$, then we have a trivial (z, v)-path. Assume that $v \neq z$. Let w be the parent of v in T. If $w \in V(F')$, then $w \prec v$ and, by the inductive hypothesis, there is a (z, w)-path in F' and it implies the existence of a (z, v)-path. Suppose that $w \notin V(F')$, i.e., $w \in W'$. Since $d_T(w) = 2$, there is $u \in Y$ such that $w = y(u, v)$. We have that $x(u) \prec v$ and there is a $(z, x(u))$-path in F' by the inductive hypothesis. It remains to observe that because $x(u)u, uv \in E(F')$, F' has a (z, v)-path as well. This concludes the proof to the claim.   ◀

Denote by R the set of the children of the vertices of $D \cup S$ in T. Observe that $|N_G(Y) \cap V(T)| \leq |D \cup S| + |R| + |U| + |W'|$. Recall that $|V(F)| \leq \ell + r$. Because F' is connected and $S \subseteq V(F')$, $|V(F')| \geq \ell$. Hence, $|W'| \leq r$. Let $q_1 = |L|$, $q_2 = |V(T) \setminus (L \cup D)|$ and $q_3 = |D|$. We have that $q_1 + 2q_2 + 3q_3 \leq 2|E(T)| = 2(q_1 + q_2 + q_3 - 1)$. Then $q_3 \leq q_1 - 2$ and $|D \cup S| \leq 2|S| - 2 = 2p - 2$, because $L \subseteq S$. Let T' be the tree obtained from T by consequent dissolving all the vertices of degree 2 that are not in S. Then $|R| \leq |E(T')| \leq 2|S| - 3 = 2p - 3$. Since $|V(T)| \geq \ell$, $|U| \leq |Y| \leq r$. We obtain that $|N_G(Y) \cap V(T)| \leq |D \cup S| + |R| + |U| + |W'| \leq 2p - 2 + 2p - 3 + r + r = 4p + 2r - 5$.   ◀

Now we are ready to prove the main result of the section.

▶ **Theorem 8.** SECLUDED STEINER TREE *can be solved in time $2^{O(p+r)} \cdot nm \cdot \log W$ by a true-biased Monte-Carlo algorithm and in time $2^{O(p+r)} \cdot nm \log n \cdot \log W$ by a deterministic algorithm for graphs with n vertices and m edges, where $r = k - \ell$ and ℓ is the size of a Steiner tree for S and W is the maximum value of ω on an input graph G.*

Proof. We construct an FPT-algorithm for SECLUDED STEINER TREE parameterized by $p + r$. The algorithm is based on the random separation techniques introduced by Cai, Chan, and Chan [6] (see also [1]). We first describe a randomized algorithm and then explain how it can be derandomized.

Let $\mathcal{I} = (G, \omega, S, k, C)$ be an instance of SECLUDED STEINER TREE, ℓ be the size of a Steiner tree for $S = \{s_1, \dots, s_p\}$ and $r = k - \ell$. Without loss of generality we assume that $p \geq 2$ and $r \geq 1$ as for $p = 1$ or $r = 0$, the problem is trivial. We also can assume that G is connected.

Description of the algorithm.   In each iteration of the algorithm we color the vertices of G independently and uniformly at random by two colors. In other words, we partition $V(G)$ into two sets R and B. We say that the vertices of R are *red*, and the vertices of B are *blue*. Our algorithm can recolor some blue vertices red, i.e., the sets R and B can be modified. Our aim is to find a connected subgraph T of G with $S \subseteq V(T)$ such that $|N_G[V(T)]| \leq k$, $\omega(N_G[V(T)]) \leq C$ and $V(T) \subseteq R$.

Step 1.   If $G[R]$ has a component H such that $S \subseteq V(H)$, then find a spanning tree T of H. If $|N_G[V(T)]| \leq k$ and $\omega(N_G[V(T)]) \leq C$, then return T and stop; otherwise, return that $\mathcal{I}$ is no-instance and stop.

Step 2.   If there is $s_i \in S$ such that $s_i \notin R$ or $N_G(s_i) \cap R = \emptyset$, then return that $\mathcal{I}$ is no-instance and stop.

Step 3.   Find a component H of $G[R]$ with $s_1 \in V(H)$. If there is a pendant vertex $u \notin S$ of H that is adjacent in G to a unique vertex $v \in B$, then find a component of $G[B]$ that contains v, recolor its vertices red and then return to Step 1. Otherwise, return that (G, S, k) is no-instance and stop.

We repeat at most $2^{O(r+p)}$ iterations. If on some iteration we obtain a yes-answer, then we return it and the corresponding solution. Otherwise, if on every iteration we get a no-answer, we return a no-answer.

Correctness of the algorithm.   It is straightforward to see that if this algorithm returns a tree T in G with $|N_G[V(T)]| \leq k$ and $\omega(N_G[V(T)]) \leq C$, then we have a solution for the considered instance of SECLUDED STEINER TREE. We show that if $\mathcal{I}$ is a yes-instance, then there is a positive constant α that does not depend on n and r such that the algorithm finds a tree T in G with $|N_G[V(T)]| \leq k$ and $\omega(N_G[V(T)]) \leq C$ with probability at least α after $2^{O(p+r)}$ executions of this algorithm for random colorings.

Suppose that $\mathcal{I}$ is a yes-instance. Then there is a tree T in G such that $S \subseteq V(T)$, $|N_G[V(T)]| \leq k$ and $\omega(N_G[V(T)]) \leq C$. Without loss of generality we assume that T is inclusion minimal. Let $F = G[V(T)]$, $X = \{v \in V(F)|d_F(v) \geq 3\} \cup S$, $X' = N_F(X)$, $Y = N_G(V(T))$ and $Y' = N_G(Y) \cap V(T)$. For each $v \in Y' \setminus S$, we arbitrarily select two distinct neighbors $z_1(v)$ and $z_2(v)$ in T. Because the leaves of T are in S, we have that v is not a leaf and thus has at least two neighbors. Let $Z = \{z_i(v)|v \in Y' \setminus S, i = 1, 2\}$. Let $W = X \cup X' \cup Y \cup Y' \cup Z$.

By Lemma 5, $|X| \leq 4p-6$ and $|X'| \leq 4p-6$. By Lemma 6, $|Y'| \leq 4p+2r-5$ and, therefore, $|Z| \leq 8p+4r-10$. Because $|V(T)| \geq \ell$ and $|N_G[V(T)]| \leq \ell+r$, we have that $|Y| \leq r$. Hence $|W| \leq |X|+|X'|+|Y|+|Y'|+|Z| \leq 4p-6+4p-6+r+4p+2r-5+8p+4r-10 = 20p+7r-27$. Let $N = 20p+7r-27$. Then with probability at least 2^{-N}, the vertices of Y are colored blue and the vertices of $X \cup X' \cup Y' \cup Z$ are colored red, i.e., $W \cap V(T) \subseteq R$ and $W \setminus V(T) \subseteq B$. The probability that for a random coloring, the vertices of W are colored incorrectly, i.e., $W \cap V(T) \cap B \neq \emptyset$ or $(W \setminus V(T)) \cap R \neq \emptyset$, is at most $1 - 2^{-N}$. Hence, if we consider 2^N random colorings, then the probability that the vertices of W are colored incorrectly for all the colorings is at most $(1 - 2^{-N})^{2^N}$, and with probability at least $1 - (1 - 2^{-N})^{2^N}$ for at least one coloring we will have $W \cap V(T) \subseteq R$ and $W \setminus V(T) \subseteq B$. Since $(1 - 2^{-N})^{2^N} \leq 1/e$, we have that $1 - (1 - 2^{-N})^{2^N} \leq 1 - 1/e$. Thus if $\mathcal{I}$ is a yes-instance, after 2^N random colorings of G, we have that at least one of the colorings is successful with a constant success probability $\alpha = 1 - 1/e$.

Assume that for a random red-blue coloring of G, $W \cap V(T) \subseteq R$ and $W \setminus V(T) \subseteq B$. We show that in this case the algorithm finds a tree T' with $S \subseteq V(T') \subseteq V(T)$. Clearly, $|N_G[V(T')]| \leq |N_G[V(T)]| \leq k$ and $\omega(N_G[V(T')]) \leq \omega(N_G[V(T)]) \leq C$ in this case.

We claim that for every connected component H of $G[R]$, either $V(H) \subseteq V(T)$ or $V(H) \cap V(T) = \emptyset$. To obtain a contradiction, assume that there are $u, v \in V(H)$ such that $u \in V(T)$ and $v \notin V(T)$. Indeed, H is connected, and thus contains an (u, v) path P. Since P goes from $V(T)$ to $v \notin V(T)$, path P should contain a vertex $w \in N_G(T) = Y$. But w is colored blue, which is a contradiction to the assumption that P is in the red component H. By the same arguments, for any component H of $G[B]$, either $V(H) \subseteq V(T)$ or $V(H) \cap V(T) = \emptyset$.

We consider Steps 1–3 of the algorithm and show their correctness.

Suppose that $G[R]$ has a component H such that $S \subseteq V(H)$. Because $S \subseteq W \cap V(T) \subseteq R$, $V(H) \cap V(T) \neq \emptyset$ and, therefore, $V(H) \subseteq V(T)$. Then for every spanning tree T' of H, $S \subseteq V(T')$ and $N_G[V(T')] \subseteq N_G[V(T)]$. Therefore, $|N_G[V(T')]| \leq |N_G[V(T)]| \leq k$ and $\omega(N_G[V(T')]) \leq \omega(N_G[V(T)]) \leq C$. Hence, if a component of $G[R]$ contains S, then we find a solution. This concludes the proof of the correctness of the first step.

Let us assume that the algorithm does not stop at Step 1. For the right coloring, because $S \subseteq X$ and $N_F(S) \subseteq X'$, for every $s_i \in S$, we have that $s_i \in R$. Moreover, because $p \geq 2$, at least one neighbor of s_i in G is in R. Thus the only reason why the algorithm stops at Step 2 is due to the wrong coloring. Consider the case when the algorithm does not stop after Step 2.

Suppose that H is a component of $G[R]$ with $s_1 \in V(H)$. Because the algorithm did not stop in Step 2, such a component H exists and has at least 2 vertices. Recall that $V(H) \subseteq V(T)$. Because we proceed in Step 1, we conclude that $S \setminus V(H) \neq \emptyset$. Then there is a vertex $u \in V(H)$ which has a neighbor v in T such that $v \in B$. If $u \in S$, then $v \in X'$, but this contradicts the assumption $X' \subseteq R$. Hence, $u \notin S$. Suppose that $d_H(u) \geq 2$. In this case $d_F(u) \geq 3$ and $v \in X'$; a contradiction. Therefore, u is a pendant vertex of H.

Let $u \notin S$ be an arbitrary pendant vertex of H. If u has no neighbors in B, then u is a leaf of T that does not belong to S but this contradicts the inclusion minimality of T. Assume that u is adjacent to at least two distinct vertices of B. Because T is an inclusion minimal tree spanning S, vertex u has at least two neighbors in T and u has a neighbor $v \in B$ in T. Let $w \in (N_G(u) \cap B) \setminus \{v\}$. If $w \in V(T)$, then $d_F(u) \geq 3$ and, therefore, $u \in X$ and $v, w \in X'$; a contradiction with $X' \subseteq R$. Hence, $w \notin V(T)$. Moreover, v is the unique neighbor of u in T that belongs to B. Then $w \in Y$ and $v \in \{z_1(u), z_2(u)\}$; a contradiction with $Z \subseteq R$. We obtain that u is adjacent in G to a unique vertex $v \in B$. Let H' be the component of $G[B]$ that contains v. Since T is an inclusion minimal tree that spans S, u has at least two neighbors in T. It implies that $v \in V(T)$, therefore $V(H') \subseteq V(T)$. We recolor the vertices of H' red in Step 3. For the new coloring the vertices of Y are blue and the vertices of $W \setminus Y$ are red. Therefore, we keep the crucial property of the considered coloring but we increase the size of the component of $G[R]$ containing s_1.

To conclude the correctness proof, it remains to observe that in Step 3 we increase the number of vertices in the component of $G[R]$ that contains s_1. Hence, after at most n repeats of Steps 1-3, we obtain a component in $G[R]$ that includes S and return a solution in Step 1.

It is straightforward to verify that each of Steps 1–3 can be done in time $O(m \log W)$. Because the number of iterations is at most n, we obtain that the total running time is $2^{O(p+r)} \cdot nm \log W$.

This algorithm can be derandomized by standard techniques (see [1, 6]). The random colorings can be replaced by the colorings induced by *universal sets*. Let n and q be positive integers, $q \leq n$. An (n, q)-*universal set* is a collection of binary vectors of length n such that for each index subset of size q, each of the 2^q possible combinations of values appears in some vector of the set. It is known that an (n, q)-universal set can be constructed in FPT-time with the parameter q. The best construction is due to Naor, Schulman and Srinivasan [20]. They obtained an (n, q)-universal set of size $2^q \cdot q^{O(\log q)} \log n$, and proved that the elements of the sets can be listed in time that is linear in the size of the set. In our case n is the number of vertices of G and $q = 20p + 7r - 27$.   ◀

We complement Theorem 8 by showing that it is unlikely that SECLUDED STEINER TREE is FPT if parameterized by r only. To show it, we use the standard reduction from the SET COVER problem (see, e.g., [19]). Notice that we prove that SECLUDED STEINER TREE is co-W[1]-hard, i.e., we show that it is W[1]-hard to decide whether we have a no-answer.

▶ **Theorem 9** (∗). SECLUDED STEINER TREE *without costs is co-*W[1]-*hard when parameterized by* r, *where* $r = k - \ell$ *and* ℓ *is the size of a Steiner tree for* S.

5 Structural parameterizations of Secluded Steiner Tree

In this section we consider different algorithmic and complexity results concerning different structural parameterizations of secluded connectivity problems. We consider parameterizations by the treewidth, size of the solution, maximum degree and the size of a vertex cover of the input graph (see [13] for definitions of these parameters.) We show that under reasonable complexity assumptions SECLUDED PATH without costs has no polynomial kernel when parameterized by $k + t + \Delta$, where t is the treewidth and Δ is the maximum degree of the input graph. We obtain the same result for the cases when the problem is parameterized by $k - \ell + t + \Delta$, where ℓ is the length of the shortest path between terminals.

▶ **Theorem 10** (∗). SECLUDED PATH *without costs on graphs of treewidth at most* t *and maximum degree at most* Δ *admits no polynomial kernel unless* NP $\subseteq$ coNP /poly *when parameterized by* $k + t + \Delta$ *or* $(k - \ell) + t + \Delta$, *where* ℓ *is the length of the shortest path between terminals.*

Observe that Theorem 10 immediately implies that SECLUDED PATH without costs has no polynomial kernel unless NP $\subseteq$ coNP /poly when parameterized by k or $k - \ell$. The next natural question is if parameterization by a stronger parameter can lead to a polynomial kernel. Let us note that the treewidth of a graph is always at most the minimum size of its vertex cover. The following theorem provides lower bounds for parameterization by the minimum size of a vertex cover.

▶ **Theorem 11.** SECLUDED PATH *without costs on graphs with the vertex cover number at most* w *has no polynomial kernel unless* NP $\subseteq$ coNP /poly *when parameterized by* w.

Proof. The proof uses the cross-composition technique introduced by Bodlaender, Jansen and Kratsch [5]. We show that the 3-SATISFIABILITY problem OR-cross composes into SECLUDED PATH without costs. Recall that 3-SATISFIABILITY asks for given boolean variables $x_1, \dots, x_n$ and clauses $C_1, \dots, C_m$ with 3 literals each, whether the formula $\phi = C_1 \wedge \dots \wedge C_m$ can be satisfied. It is well-known that 3-SATISFIABILITY is NP-complete [16]. We assume that two instances of 3-SATISFIABILITY are equivalent if they have the same number of variables and the same number of clauses.

Consider t equivalent instances of 3-SATISFIABILITY with the same boolean variables $x_1, \dots, x_n$ and the sets of clauses $\mathcal{C}_i = \{C_1^i, \dots, C_m^i\}$ for $i \in \{1, \dots, t\}$. Without loss of generality we assume that $t = \binom{2q}{q}$ for a positive integer q; otherwise, we add minimum number of copies of $\mathcal{C}_1$ to get this property. Notice that $\binom{2q}{q} = \Theta(4^q / \sqrt{\pi q})$ and $q = O(\log t)$. Let $I_1, \dots, I_t$ be pairwise distinct subsets of $\{1, \dots, 2q\}$ of size q. Notice that each $i \in \{1, \dots, 2q\}$ is included exactly in $d = \binom{2q-1}{q-1}$ sets. Let $k = (q + 3d)m + 3q + 4n + 2$. We construct the graph G as follows (see Fig. 1).

(i) Construct $n + 1$ vertices $u_0, \dots, u_n$. Let $s_1 = u_0$.

(ii) For each $i \in \{1, \dots, n\}$, construct vertices $x_i, y_i, \overline{x}_i, \overline{y}_i$ and edges $u_{i-1}y_i, y_iu_i, y_ix_i$, and $u_{i-1}\overline{y}_i, \overline{y}_iu_i, \overline{y}_i\overline{x}_i$.

(iii) For each $j \in \{0, \dots, m\}$, construct a set of vertices $W_j = \{w_1^j, \dots, w_{2q}^j\}$.

(iv) Construct a vertex s_2 and edges $u_nw_1^0, \dots, u_nw_{2q}^0$ and $w_1^ms_2, \dots, w_{2q}^ms_2$.

(v) For each $j \in \{1, \dots, m\}$ and $h \in \{1, \dots, t\}$,

- construct 3 vertices $c_{jh}^1, c_{jh}^2, c_{jh}^3$;
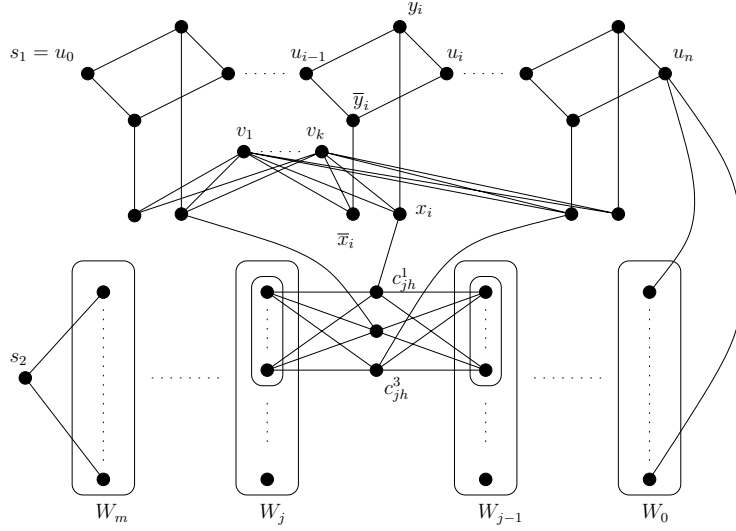
Figure 1 Construction of G.

- construct edges $c_{jh}^1 w_r^{j-1}, c_{jh}^2 w_r^{j-1}, c_{jh}^3 w_r^{j-1}$ and $c_{jh}^1 w_r^j, c_{jh}^2 w_r^j, c_{jh}^3 w_r^j$ for all $r \in I_h$;
- consider the clause $C_j^h = (z_1 \vee z_2 \vee z_3)$ and for $l \in \{1, 2, 3\}$, construct an edge $c_{jh}^l x_i$ if $z_l = x_i$ for some $i \in \{1, \dots, n\}$ and construct an edge $c_{jh}^l \overline{x_i}$ if $z_l = \overline{x_i}$.

(vi) Construct k vertices $v_1, \dots, v_k$ and edges $x_i v_l, \overline{x_i} v_l$ for $i \in \{1, \dots, n\}$ and $l \in \{1, \dots, k\}$. Observe that the set of vertices

$$X = (\cup_{i=1}^n \{x_i, y_i, \overline{x_i}, \overline{y_i}\}) \cup (\cup_{j=0}^m W_j)$$

is a vertex cover in G of size $4n + 2q(m+1) = O(n + m \log t)$.

We show that G has an (s_1, s_2)-path P with $|N_G[V(P)]| \le k$ if and only if there is $h \in \{1, \dots, t\}$ such that $x_1, \dots, x_n$ have a truth assignment satisfying all the clauses of $\mathcal{C}_h$.

Suppose that $x_1, \dots, x_n$ have an assignment that satisfies all the clauses of $\mathcal{C}_h$. First, we construct the (s_1, u_n)-path P' by the concatenation of the following paths: for each $i \in \{1, \dots, n\}$, we take the path $u_{i-1} y_i u_i$ if $x_i = true$ in the assignment and we take $u_{i-1} \overline{y_i} u_i$ if $x_i = false$. Let $r \in I_h$. We construct the (w_r^0, w_r^m)-path P'' by concatenating $w_r^{j-1} c_{jh}^{l_j} w_r^j$ for $j \in \{1, \dots, m\}$ where $l_j \in \{1, 2, 3\}$ is chosen as follows. Each clause $C_j^h = z_1 \vee z_2 \vee z_3 = true$ for the assignment, i.e., $z_l = true$ for some $l \in \{1, 2, 3\}$; we set $l_j = l$. Finally, we set $P = P' + u_n w_h^0 + P'' + w_h^m s_2$. It is straightforward to verify that $|N_G[V(P)]| = k$.

Suppose now that there is an (s_1, s_2)-path in G with $|N_G[V(P)]| \le k$. We assume that P is an induced path. Observe that $x_i, \overline{x_i} \notin V(P)$ for $i \in \{1, \dots, n\}$, because $d_G(x_i), d_G(\overline{x_i}) > k$. Therefore, P has an (s_1, u_n)-subpath P' such that $u_0, \dots, u_n \in V(P')$ and for each $i \in \{1, \dots, n\}$, either $y_i \in V(P')$ or $\overline{y_i} \in V(P')$. We set the variable $x_i = true$ if $x_i \in V(P')$ and $x_i = false$ otherwise. We show that this truth assignment satisfies all the clauses of some $\mathcal{C}_h$.

Observe that $|N_G[V(P')]| = 4n + 2q + 1$. Clearly, $s_2 \in V(P)$. Notice also that P has at least one vertex in each W_j for $j \in \{0, \dots, m\}$, and for each $j \in \{1, \dots, m\}$, at least one vertex among the vertices c_{jh}^l for $h \in \{1, \dots, t\}$ and $l \in \{1, 2, 3\}$ is in P. For each $j \in \{1, \dots, m\}$, any two verices $w_r^{j-1} \in W_{j-1}$ and $w_{r'}^j \in W_j$ have at least $3d$ neighbors among the vertices c_{jf}^l for $f \in \{1, \dots, t\}$ and $l \in \{1, 2, 3\}$. Moreover, if $r \ne r'$, they have at least $3d + 6$ such neighbors, because there are two subsets $I, I' \subseteq \{1, \dots, 2q\}$ of size q such

that $r \in I \setminus I'$ and $r' \in I' \setminus I$. For each $j \in \{1, \ldots, m-1\}$, any two vertices c_{jh}^l and $c_{j+1\ h'}^{l'}$ for $h, h' \in \{1, \ldots, t\}$ and $l, l' \in \{1, 2, 3\}$ have at least q neighbors in W_j. Moreover, if $h \neq h'$, they have at least $q + 2$ such neighbors, because $|I_h \cup I_{h'}| \geq q + 2$. Taking into account that $d_G(s_2) = 2q$, we obtain that

$$k \geq |N_G[V(P)]| \geq |N_G[V(P')]| + 3dm + q(m-1) + 2q + 1 = k.$$

It implies that P has exactly one vertex in each W_j for $j \in \{0, \ldots, m\}$, and for each $j \in \{1, \ldots, m\}$, exactly one vertex among the vertices c_{jh}^l for $h \in \{1, \ldots, t\}$ and $l \in \{1, 2, 3\}$ is in P. Moreover, there is $r \in \{1, \ldots, 2q\}$ and $h \in \{1, \ldots, t\}$ such that $w_r^j \in V(P)$ and $c_{jh}^{l_j} \in V(P)$ for $j \in \{0, \ldots, m\}$ and $l_j \in \{1, 2, 3\}$. We claim that all the clauses of $\mathcal{C}_r$ are satisfied. Otherwise, if there is a clause $C_j^r = (z_1 \vee z_2 \vee z_3)$ that is not satisfied, then the neighbors of $c_{jh}^1, c_{jh}^2, c_{jh}^3$ among the vertices $x_i, \overline{x}_i$ for $i \in \{1, \ldots, n\}$ are not in $N_G[V(P')]$. It immediately implies that $|N_G[V(P)]| > k$; a contradiction. ◀

However, if we consider even stronger parameterization, by vertex cover number and by the size of the solution, then we obtain the following theorem.

▶ **Theorem 12** (∗)**.** *The* Secluded Steiner Tree *problem admits a kernel with at most* $2w(k+1)$ *vertices on graphs with the vertex cover number at most* w.

Recall that Chechik et al. [8] showed that if the treewidth of the input graph does not exceed t, then the Secluded Steiner Tree problem is solvable in time $2^{\mathcal{O}(t \log t)} \cdot n^{\mathcal{O}(1)} \cdot \log W$, where W is the maximum value of ω on an input graph G. We observe that the running time could be improved by applying modern techniques for dynamic programming over tree decompositions proposed by Cygan et al. [10], Bodlaender et al. [3] and Fomin et al. [14]. Essentially, the algorithms for Secluded Steiner Tree are constructed along the same lines as the algorithms for Steiner Tree described in [10, 3, 14]. Hence, for simplicity, we only sketch the randomized algorithm based on the Cut&Count technique introduced by Cygan et al. [10] for Secluded Steiner Tree without costs in this conference version of our paper.

▶ **Theorem 13** (∗)**.** *There is a true-biased Monte Carlo algorithm solving the* Secluded Steiner Tree *without costs in time* $4^t \cdot n^{\mathcal{O}(1)}$, *given a tree decomposition of width at most* t.

The algorithm based on the Cut&Count technique can be generalized for Secluded Steiner Tree with costs in the same way as the algorithm for Steiner Tree in [10]. This way we can obtain the algorithm that runs in time $4^t \cdot (n + W)^{\mathcal{O}(1)}$ where W is the maximal cost of vertices. One can obtain a deterministic algorithm and improve the dependence on W using the representative set technique for dynamic programming over tree decompositions introduced by Fomin et al. [14]. Again by the same approach as for Steiner Tree, it is possible to solve Secluded Steiner Tree deterministically in time $\mathcal{O}((2 + 2^{\omega+1})^t \cdot (n + \log W)^{\mathcal{O}(1)})$ (here ω is the matrix multiplication constant). We postpone the proof till the full version of the paper.

─── **References** ───

1 Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *J. ACM*, 42(4):844–856, 1995.
2 Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Fourier meets möbius: fast subset convolution. In *STOC 2007*, pages 67–74. ACM, 2007.

3   Hans L. Bodlaender, Marek Cygan, Stefan Kratsch, and Jesper Nederlof. Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth. In *ICALP 2013, Part I*, volume 7965 of *Lecture Notes in Computer Science*, pages 196–207. Springer, 2013.

4   Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin. On problems without polynomial kernels. *J. Comput. Syst. Sci.*, 75(8):423–434, 2009.

5   Hans L. Bodlaender, Bart M. P. Jansen, and Stefan Kratsch. Kernelization lower bounds by cross-composition. *SIAM J. Discrete Math.*, 28(1):277–305, 2014.

6   Leizhen Cai, Siu Man Chan, and Siu On Chan. Random separation: A new method for solving fixed-cardinality optimization problems. In *IWPEC 2006*, volume 4169 of *Lecture Notes in Computer Science*, pages 239–250. Springer, 2006.

7   Shiri Chechik, Matthew P. Johnson, Merav Parter, and David Peleg. Secluded connectivity problems. *CoRR*, abs/1212.6176, 2012.

8   Shiri Chechik, Matthew P. Johnson, Merav Parter, and David Peleg. Secluded connectivity problems. In *ESA 2013*, volume 8125 of *Lecture Notes in Computer Science*, pages 301–312. Springer, 2013.

9   Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms.* Springer, 2015.

10 Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michał Pilipczuk, Johan M. M. van Rooij, and Jakub Onufry Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. In *FOCS 2011*, pages 150–159. IEEE, 2011.

11 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity.* Texts in Computer Science. Springer, 2013.

12 S. E. Dreyfus and R. A. Wagner. The Steiner problem in graphs. *Networks*, 1(3):195–207, 1971.

13 Fedor V. Fomin, Petr A. Golovach, Nikolay Karpov, and Alexander S. Kulikov. Parameterized complexity of secluded connectivity problems. *CoRR*, abs/1502.03989, 2015.

14 Fedor V Fomin, Daniel Lokshtanov, and Saket Saurabh. Efficient computation of representative sets with applications in parameterized and exact algorithms. In *SODA 2014*, pages 142–151, 2014.

15 Jianhang Gao, Qing Zhao, and Ananthram Swami. The thinnest path problem for secure communications: A directed hypergraph approach. In *Proceedings of the 50th Annual Allerton Conference on Communication, Control, and Computing, 2012*, pages 847–852. IEEE, 2012.

16 M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

17 Alexander Gilbers. *Visibility Domains and Complexity.* PhD thesis. Rheinische Friedrich-Wilhelms-Universität Bonn, Bonn, 2013.

18 Matthew P. Johnson, Ou Liu, and George Rabanca. Secluded path via shortest path. In *SIROCCO 2014*, volume 8576 of *Lecture Notes in Computer Science*, pages 108–120. Springer, 2014.

19 Richard M. Karp. Reducibility among combinatorial problems. In *Proceedings of a symposium on the Complexity of Computer Computations*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972.

20 M. Naor, L.J. Schulman, and A. Srinivasan. Splitters and near-optimal derandomization. In *FOCS 1995*, pages 182–191. IEEE, 1995.

21 Jesper Nederlof. Fast polynomial-space algorithms using inclusion-exclusion. *Algorithmica*, 65(4):868–884, 2013.