Finding Connected Secluded Subgraphs[⋆,⋆⋆]

Petr A. Golovach[a,∗], Pinar Heggernes[a], Paloma T. Lima[a], Pedro Montealegre[b]

[a]*Department of Informatics, University of Bergen, N-5020 Bergen, Norway*
[b]*Facultad de Ingeniería y Ciencias, Universidad Adolfo Ibáñez, Santiago, Chile*

Abstract

Problems related to finding induced subgraphs satisfying given properties form one of the most studied areas within graph algorithms. However, for many applications, it is desirable that the found subgraph has as few connections to the rest of the graph as possible, which gives rise to the Secluded Π-Subgraph problem. Here, input k is the size of the desired subgraph, and input t is a limit on the number of neighbors this subgraph has in the rest of the graph. This problem has been studied from a parameterized perspective, and unfortunately it turns out to be W[1]-hard for many graph properties Π, even when parameterized by $k + t$. We show that the situation changes when we are looking for a connected induced subgraph satisfying Π. In particular, we show that the Connected Secluded Π-Subgraph problem is FPT when parameterized by just t for many important graph properties Π.

Keywords: secluded subgraph, parameterized complexity, forbidden subgraphs

1. Introduction

Vertex deletion problems are central in parameterized algorithms and complexity, and they have contributed hugely to the development of new algorithmic methods. The Π-Deletion problem, with input a graph G and an integer ℓ, asks whether at most ℓ vertices can be deleted from G so that the resulting graph satisfies graph property Π. Its dual, the Π-Subgraph problem, with input G and k, asks whether G contains an induced subgraph on at least k vertices satisfying Π. The problems were introduced already in 1980 by Yannakakis and

⋆The preliminary version of this paper appeared as an extended abstract in the proceedings of IPEC 2017.

⋆⋆This work is supported by the Research Council of Norway by the project CLASSIS (grant no. 249994) and and by National Commission for Scientific and Technological Research of Chile via PAI + PAI77170068.

∗Corresponding author. University of Bergen, PB 7803, N-5020, Bergen, Norway. Phone: +47 55584385

Email addresses: petr.golovach@uib.no (Petr A. Golovach), pinar.heggernes@uib.no (Pinar Heggernes), paloma.lima@uib.no (Paloma T. Lima), p.montealegre@uai.cl (Pedro Montealegre)

Lewis [15], who showed their NP-completeness for almost all interesting graph properties Π. During the last couple of decades, these problems have been studied extensively with respect to parameterized complexity and kernelization, which has resulted in numerous new techniques and methods in these fields [6, 8].

In many network problems, the size of the *boundary* between the subgraph that we are looking for and the rest of the graph makes a difference. A small boundary limits the exposure of the found subgraph, and notions like isolated cliques have been studied in this respect [11, 12, 14]. Several measures for the boundary have been proposed; in this work we use the open neighborhood of the returned induced subgraph. For a set of vertices U of a graph G and a nonnegative integer t, we say that U is *t-secluded* if $|N_G(U)| \leq t$. Analogously, an induced subgraph H of G is *t-secluded* if the vertex set of H is t-secluded. For a given graph property Π, we get the following formal definition of the problem Secluded Π-Subgraph.

Secluded Π-Subgraph

Input:	A graph G and nonnegative integers k and t.
Task:	Decide whether G contains a t-secluded induced subgraph H on at least k vertices, satisfying Π.

Lewis and Yannakakis [15] showed that Π-Subgraph is NP-complete for every hereditary nontrivial graph property Π. This immediately implies that Secluded Π-Subgraph is NP-complete for every such Π. As a consequence, the interest has shifted towards the parameterized complexity of the problem, which has been studied by van Bevern et al. [1] for several classes of properties Π. Unfortunately, in most cases Secluded Π-Subgraph proves to be W[1]-hard, even when parameterized by $k + t$. In particular, it is W[1]-hard to decide whether a graph G has a t-secluded independent set of size k when the problem is parameterized by $k + t$ [1]. We show that the situation changes when the secluded subgraph we are looking for is required to be connected, in which case we are able to obtain positive results that apply to many properties Π. In fact, connectivity is central in recently studied variants of secluded subgraphs, like Secluded Path [4, 13, 17, 2] (see also [21]) and Secluded Steiner Tree [9]. However, in these problems, the measure is the size of the closed neighborhood of a path or a tree connecting a given set of vertices. The following formal definition describes the Connected Secluded Π-Subgraph problem that we study. For generality, we define a weighted problem.

Connected Secluded Π-Subgraph

Input:	A graph G, a weight function $\omega\colon V(G) \to \mathbb{Z}_{>0}$, a nonnegative integer t and a positive integer w.
Task:	Decide whether G contains a connected t-secluded induced subgraph H with $\omega(V(H)) \geq w$, satisfying Π.

Observe that CONNECTED SECLUDED Π-SUBGRAPH remains NP-complete for all hereditary graph properties Π such that the property combining Π and connectivity is nontrivial, following the results of Yannakakis [22]. It can also be seen that CONNECTED SECLUDED Π-SUBGRAPH parameterized by w is W[1]-hard even for unit weights, if it is W[1]-hard with parameter k to decide whether G has a connected induced subgraph on at least k vertices, satisfying Π (see, e.g., [8, 20]).

It is thus more interesting to consider parameterization by t, and we show that CONNECTED SECLUDED Π-SUBGRAPH is fixed-parameter tractable (FPT) when parameterized by t for many important graph properties Π. Our main result is given in Section 4 where we consider CONNECTED SECLUDED Π-SUBGRAPH for all graph properties Π that are characterized by finite sets $\mathcal{F}$ of forbidden induced subgraphs and refer to this variant of the problem as CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH. We show that the problem is FPT when parameterized by t by proving the following theorem.

Theorem 1. CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH *can be solved in time* $2^{2^{2^{\mathcal{O}(t\log t)}}}\cdot n^{\mathcal{O}(1)}$.

We prove the theorem by making use of the *recursive understanding* technique introduced by Chitnis et al. [5] for graph problems. This technique is based on the following idea. Suppose that the input graph has a separator of bounded size that divides the graph into two sufficiently big parts. Then we solve the problem recursively for one of the parts and replace this part by an equivalent graph such that the replacement keeps all essential (partial) solutions of the original part. By such a replacement we obtain a graph of smaller size and this allows to use recursion. Otherwise, if there is no separator of bounded size separating the graph into two big parts, then the graph is said to be *unbreakable* and we exploit this property to solve the problem directly. It was shown by Chitnis et al. [5] that this technique provides a powerful tool for obtaining FPT algorithms for cut problems. This technique was further developed in [7, 16]. In particular, very recently, Lokshtanov et al. [16] showed the meta-algorithmic theorem that roughly states that if a parameterized problem is FPT on unbreakable graphs and can be expressed in *Counting Monadic Second-Order Logic* (CMSO), then the problem is FPT for general graphs (we refer to [16] for the precise statements and the definition of CMSO). Nevertheless, it should be noted that this theorem is nonconstructive and provides only an existential result. Note also that we still have to go through the unbreakable case to apply the theorem. Since solving CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH is already nontrivial on unbreakable graphs and we believe that it would be interesting to obtain a constructive algorithm with explicit running time, we prefer to apply the recursive understanding technique directly.

Further, in Section 5 we show that we can get faster algorithms for CONNECTED SECLUDED Π-SUBGRAPH when Π is the property of being a complete graph, a star, a d-regular graph, and a path. Finally, in Section 6 we briefly discuss extensions of our results for some other properties Π and kernelization for CONNECTED SECLUDED Π-SUBGRAPH.

2. Preliminaries

We consider only finite undirected simple graphs. We use n to denote the number of vertices and m the number of edges of the considered graphs unless it creates confusion. A graph G is identified by its vertex set $V(G)$ and edge set $E(G)$. For $U \subseteq V(G)$, we write $G[U]$ to denote the subgraph of G induced by U. We write $G - U$ to denote the graph $G[V(G) \setminus U]$; for a single-element $U = \{u\}$, we write $G - u$. A set of vertices U is *connected* if $G[U]$ is a connected graph. For a vertex v, we denote by $N_G(v)$ the *(open) neighborhood* of v in G, i.e., the set of vertices that are adjacent to v in G. For a set $U \subseteq V(G)$, $N_G(U) = (\bigcup_{v \in U} N_G(v)) \setminus U$. We denote by $N_G[v] = N_G(v) \cup \{v\}$ the *closed neighborhood* of v; respectively, $N_G[U] = \bigcup_{v \in U} N_G[v]$. The *degree* of a vertex v is $d_G(v) = |N_G(v)|$. Two vertices u and v of graph G are *true twins* if $N_G[u] = N_G[v]$, and *false twins* if $N_G(u) = N_G(v)$. A set of vertices $S \subset V(G)$ of a connected graph G is a *(vertex) separator* if $G - S$ is disconnected. A vertex v is a *cut vertex* if $\{v\}$ is a separator. For two sets of vertices $A, B \subseteq V(G)$, a set $S \subseteq V(G)$ is an (A, B)*-separator* if $G - S$ has no path with one end-vertex in A and the other in B; S is an (inclusion) minimal separator if there is no proper subset of S that is an (A, B)-separator. A pair (A, B), where $A, B \subseteq V(G)$ and $A \cup B = V(G)$, is a *separation of G of order $|A \cap B|$* if G has no edge uv with $u \in A \setminus B$ and $v \in B \setminus A$, i.e., $A \cap B$ is an (A, B)-separator.

A graph property is *hereditary* if it is preserved under vertex deletion, or equivalently, under taking induced subgraphs. A graph property is *trivial* if either the set of graphs satisfying it, or the set of graphs that do not satisfy it, is finite. Let F be a graph. We say that a graph G is *F-free* if G has no induced subgraph isomorphic to F. For a set of graphs $\mathcal{F}$, a graph G is *$\mathcal{F}$-free* if G is F-free for every $F \in \mathcal{F}$. Let Π be the property of being $\mathcal{F}$-free. Then, depending on whether $\mathcal{F}$ is a finite or an infinite set, we say that Π is *characterized by a finite / infinite set of forbidden induced subgraphs.*

3. Connected Secluded Π-Subgraph parameterized by $k + t$

In this section, we consider the special colored variant of Connected Secluded Π-Subgraph and show that the problem is FPT when parameterized by $k + t$. We will rely on this result in the subsequent sections, however we believe that it is also of interest on its own.

We say that a mapping $c \colon V(G) \to \mathbb{N}$ is a *coloring* of G; note that we do not demand a coloring to be proper. Analogously, we say that Π is a *property of colored graphs* if Π is a property on pairs (G, c), where G is a graph and c is a coloring. Moreover, we allow Π be a property of a subgraph with respect to a colored graph. Notice that if some vertices of a graph have labels, then we can assign to each label (or a combination of labels if a vertex can have several labels) a specific color and assign some color to unlabeled vertices. Then we can redefine a considered graph property with the conditions imposed by labels as a property of colored graphs. For a property Π of colored graphs, we define the following problem.

Connected Secluded Colored Π-Subgraph of Exact Size

Input:	A graph G, coloring $c\colon V(G) \to \mathbb{N}$, a weight function $\omega\colon V(G) \to \mathbb{Z}_{\geq 0}$ and nonnegative integers k, t and w.
Task:	Decide whether G contains a connected t-secluded induced subgraph H such that (H, c'), where $c' = c|_{V(H)}$, satisfies Π, $|V(H)| = k$ and $\omega(V(H)) \geq w$.

Observe that we allow zero weights. We give two algorithms for Connected Secluded Colored Π-Subgraph of Exact Size with different running times. The first algorithm is based on Lemmas 3.1 and 3.2 of Fomin and Villanger [10], which we summarize in Lemma 1 below. The second algorithm uses Lemma 2 by Chitnis et al. [5], and we are going to use it when $k \gg t$.

Lemma 1 ([10])**.** *Let G be a graph. For every $v \in V(G)$ and $k, t \geq 0$, the number of connected vertex subsets $U \subseteq V(G)$ such that $v \in U$, $|U| = k$, and $|N_G(U)| = t$, is at most $\binom{k+t}{t}$. Moreover, all such subsets for all $v \in V(G)$ can be enumerated in time $\mathcal{O}(\binom{k+t}{t} \cdot (n+m) \cdot t \cdot (k+t))$.*

Lemma 2 ([5])**.** *Given a set U of size n and integers $0 \leq a, b \leq n$, one can construct in time $2^{\mathcal{O}(\min\{a,b\} \log(a+b))} n \log n$ a family $\mathcal{S}$ of at most $2^{\mathcal{O}(\min\{a,b\} \log(a+b))} \log n$ subsets of U such that the following holds: for any sets $A, B \subseteq U$, $A \cap B = \emptyset$, $|A| \leq a$, $|B| \leq b$, there exists a set $S \in \mathcal{S}$ with $A \subseteq S$ and $B \cap S = \emptyset$.*

Theorem 2. *If property Π can be recognized in time $f(n)$, then* Connected Secluded Colored Π-Subgraph of Exact Size *can be solved both in time $2^{k+t} \cdot f(k) \cdot n^{\mathcal{O}(1)}$, and in time $2^{\mathcal{O}(\min\{k,t\} \log(k+t))} \cdot f(k) \cdot n^{\mathcal{O}(1)}$.*

Proof. Let (G, c, ω, k, t, w) be an instance of Connected Secluded Colored Π-Subgraph of Exact Size.

First, we use Lemma 1 and in time $2^{k+t} \cdot n^{\mathcal{O}(1)}$ enumerate all connected $U \subseteq V(G)$ with $|U| = k$ and $|N_G(U)| \leq t$. By Lemma 1, we have at most $\binom{k+t}{t} tn$ sets. For every such a set U, we check in time $f(k)$ whether the given colored induced subgraph $H = G[U]$ satisfies Π and check in time $\mathcal{O}(k)$ whether $\omega(U) \geq w$. It is straightforward to see that (G, c, ω, k, t, w) is a yes-instance if and only if we find U with these properties.

To construct the second algorithm, assume that (G, c, ω, k, t, w) is a yes-instance. Then there is $U \subseteq V(G)$ such that U is a connected k-vertex set such that $|N_G(U)| \leq t$, $\omega(U) \geq w$ and the colored graph $H = G[U]$ satisfies Π. Using Lemma 2, we can construct in time $2^{\mathcal{O}(\min\{k,t\} \log(k+t))} \cdot n^{\mathcal{O}(1)}$ a family $\mathcal{S}$ of at most $2^{\mathcal{O}(\min\{k,t\} \log(k+t))} \log n$ subsets of $V(G)$ such that the following holds: for any sets $A, B \subseteq V(G)$, $A \cap B = \emptyset$, $|A| \leq k$, $|B| \leq t$, there exists a set $S \in \mathcal{S}$ with $A \subseteq S$ and $B \cap S = \emptyset$. In particular, we have that there is $S \in \mathcal{S}$ such that $U \subseteq S$ and $N_G(U) \cap S = \emptyset$. It implies that $G[U]$ is a component of $G[S]$.

Therefore, (G, c, ω, k, t, w) is a yes-instance if and only if there is $S \in \mathcal{S}$ such that a component of $G[S]$ is a solution for the instance. We construct the described set $\mathcal{S}$. Then for every $S \in \mathcal{S}$, we consider the components of $G[S]$, and for every component H, we verify in time $f(k) + \mathcal{O}(k)$, whether H gives us a solution. □

Theorem 2 immediately gives the following corollary.

Corollary 1. *If* Π *can be recognized in polynomial time, then* Connected Secluded Colored Π-Subgraph of Exact Size *can be solved both in time* $2^{k+t} \cdot n^{\mathcal{O}(1)}$*, and in time* $2^{\mathcal{O}(\min\{k,t\}\log(k+t))} \cdot n^{\mathcal{O}(1)}$.

4. Connected Secluded Π-Subgraph for properties characterized by finite sets of forbidden induced subgraphs

In this section we prove Theorem 1 and show that Connected Secluded Π-Subgraph is FPT parameterized by t when Π is characterized by a finite set of forbidden induced subgraphs. We refer to this restriction of our problem as Connected Secluded $\mathcal{F}$-Free Subgraph. Throughout this section, we assume that we are given a fixed finite set $\mathcal{F}$ of graphs.

We apply the recursive understanding technique introduced by Chitnis et al. [5]. To do this, we introduce some specific notions and obtain auxiliary results in Subsection 4.1. In particular, we introduce a special version of Connected Secluded $\mathcal{F}$-Free Subgraph tailored for the recursion and give a high level overview of our algorithm. Further, in Subsection 4.2, we consider the base of recursion. Then, in Subsection 4.3, we discuss the main recursive step of our algorithm.

4.1. Recursive understanding for Connected Secluded $\mathcal{F}$-Free Subgraph

The main idea of the recursive understanding technique [5] is to separate the input graph into two sufficiently big parts, obtain solutions for one of the parts and then use this information to reduce the graph and recurse. To apply this scheme, we first have to formalize the considered separations.

Let G be a graph. Let us remind that a pair (U, W), where $U, W \subseteq V(G)$ and $U \cup W = V(G)$, is a *separation of* G *of order* $|U \cap W|$ if G has no edge uv with $u \in U \setminus W$ and $v \in W \setminus U$, i.e., $U \cap W$ is a (U, W)-separator.

Definition 1. *Let* q *and* k *be nonnegative integers. A graph* G *is* (q, k)-unbreakable *if for every separation* (U, W) *of* G *of order at most* k*,* $|U \setminus W| \leq q$ *or* $|W \setminus U| \leq q$*.*

Combining Lemmas 19, 20 and 21 of [5], we obtain the following lemma that allows either to find a separation of a graph or decide that the graph is unbreakable.

Lemma 3 ([5])**.** *Let* q *and* k *be nonnegative integers. There is an algorithm with running time* $2^{\mathcal{O}(\min\{q,k\}\log(q+k))} \cdot n^3 \log n$ *that, for a graph* G*, either finds a separation* (U, W) *of order at most* k *such that* $|U \setminus W| > q$ *and* $|W \setminus U| > q$*, or correctly reports that* G *is* $((2q+1)q \cdot 2^k, k)$*-unbreakable.*

In our algorithm, we choose appropriate values of q and k that depend on the parameter t of the considered instance of Connected Secluded $\mathcal{F}$-Free Subgraph and verify whether the input graph is unbreakable. Assume that we obtain a separation (U, W) of order at most k such that $|U \setminus W| > q$ and $|W \setminus U| > q$. Then $U \cap W$ is a separator of G. Our aim is to deal with $G[U]$ and $G[W]$ separately and apply our algorithm recursively. For this, we have to keep track of the common vertices of $G[U]$ and $G[W]$, that is, of the vertices of $U \cap W$. These vertices form the boundary of $G[U]$ and $G[W]$ and are used to glue partial solutions. Note that in recursive steps we may cut $G[U]$ or $G[W]$ further and obtain new boundary vertices. Therefore, we have to switch to considering graphs with given sets of marked boundary vertices.

Definition 2. *Let p be a nonnegative integer. A pair (G, x), where G is a graph and $x = (x_1, \ldots, x_p)$ is a p-tuple of distinct vertices of G, is called a p-*boundaried graph *or simply a* boundaried graph*; $x = (x_1, \ldots, x_p)$ is a* boundary*. We say that (G, x) is a* properly p-boundaried graph *if each component of G has at least one vertex of the boundary.*

Note that a boundary is an ordered set. Hence, two p-boundaried graphs that differ only by the order of the vertices in their boundaries are distinct. Observe also that a boundary could be empty. When there is no ambiguity, we may omit the boundary x and say that G is a (p-) boundaried graph assuming that a boundary is given.

To deal with boundaried graphs, we need the following definitions.

Definition 3. *Two p-boundaried graphs $(G_1, x^{(1)})$ and $(G_2, x^{(2)})$, where $x^{(h)} = (x_1^{(h)}, \ldots, x_p^{(h)})$ for $h \in \{1, 2\}$, are* isomorphic *if there is an isomorphism of G_1 to G_2 that maps each $x_i^{(1)}$ to $x_i^{(2)}$ for each $i \in \{1, \ldots, p\}$.*

Definition 4. *We say that p-boundaried graphs $(G_1, x^{(1)})$ and $(G_2, x^{(2)})$ are* boundary-compatible *and denote it $(G_1, x^{(1)}) \simeq_{\mathrm{b}} (G_2, x^{(2)})$ if for any distinct $i, j \in \{1, \ldots, p\}$, $x_i^{(1)} x_j^{(1)} \in E(G_1)$ if and only if $x_i^{(2)} x_j^{(2)} \in E(G_2)$.*

Definition 5. *Let $(G_1, x^{(1)})$ and $(G_2, x^{(2)})$ be boundary-compatible p-boundaried graphs and let $x^{(h)} = (x_1^{(h)}, \ldots, x_p^{(h)})$ for $h \in \{1, 2\}$. We define the* boundary sum $(G_1, x^{(1)}) \oplus_{\mathrm{b}} (G_2, x^{(2)})$ *(or simply $G_1 \oplus_{\mathrm{b}} G_2$) as the (nonboundaried) graph obtained by taking disjoint copies of G_1 and G_2 and identifying $x_i^{(1)}$ and $x_i^{(2)}$ for every $i \in \{1, \ldots, p\}$.*

Definition 6. *Let G be a graph and let $y = (y_1, \ldots, y_p)$ be a p-tuple of vertices of G. For an s-boundaried graph (H, x) with $x = (x_1, \ldots, x_s)$ with $s \leq p$ and a subtuple $y' = (y_{i_1}, \ldots, y_{i_s})$ of y, we say that (H, x) (or simply H) is an* induced boundaried subgraph of (G, y) (or G) with respect to y' *if G contains an induced subgraph H' isomorphic to H with an isomorphism of H to H' that maps x_j to y_{i_j} for all $j \in \{1, \ldots, s\}$ and $V(H') \cap \{y_1, \ldots, y_p\} = \{y_{i_1}, \ldots, y_{i_s}\}$ (see Fig. 1 for an illustration).*

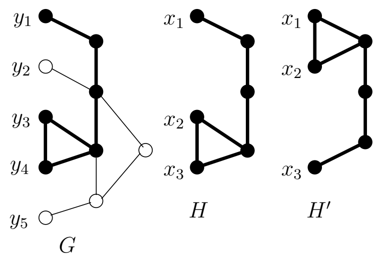

Figure 1: The 3-boundaried graph $(H,(x_1,x_2,x_3))$ is an induced boundaried subgraph of $(G,(y_1,y_2,y_3,y_4))$ with respect to (y_1,y_3,y_4). Note that $(H',(x_1,x_2,x_3))$ in not an induced boundaried subgraph of $(G,(y_1,y_2,y_3,y_4))$ with resect to any 3-subtuple of (y_1,y_2,y_3,y_4,y_5).

Recall that the goal of Connected Secluded $\mathcal{F}$-Free Subgraph is to find a connected t-secluded induced subgraph H of the input graph that, in particular, has no induced subgraph from $\mathcal{F}$. When the input graph is separated, H may become separated as well. This means that to ensure that H is $\mathcal{F}$-free, we have to verify that the parts of H do not contain induced subgraphs that together form a forbidden subgraph. To deal with this issue, we have to consider boundaried subgraphs of the graphs of $\mathcal{F}$.

Definition 7. *The set of boundaried graphs $\mathcal{F}_{\rm b}$ is the set of all pairwise non-isomorphic boundaried graphs of the form $(F[A],x)$ where $F\in\mathcal{F}$, (A,B) is separation of F, and x is a p-tuple of the vertices $A\cap B$ for $p=|A\cap B|$.*

Suppose that (G,x) is a p-boundaried graph that occurs at some step of our recursion. If H is a solution for the original instance of Connected Secluded $\mathcal{F}$-Free Subgraph, then $H=H_1\oplus_{\rm b} H_2$ where H_1 is the part of the solution inside G and H_2 is the outside part. Note that the outside parts are unknown to us when we consider (G,x). The idea to deal with this issue is to represent all possible outside parts by boundaried subgraphs of bounded size that behave differently with respect to $\mathcal{F}_{\rm b}$. For this, we introduce the following equivalence relation on the set of p-boundaried subgraphs.

Definition 8. *We say that two properly p-boundaried graphs $(G_1,x^{(1)})$ and $(G_2,x^{(2)})$, where $x^{(h)}=(x_1^{(h)},\dots,x_p^{(h)})$ for $h\in\{1,2\}$, are* equivalent (with respect to $\mathcal{F}_{\rm b}$) *and write if*

(i) $(G_1,x^{(1)})\simeq_{\rm b}(G_2,x^{(2)})$,

(ii) for any $i,j\in\{1,\dots,p\}$, $x_i^{(1)}$ and $x_j^{(1)}$ are in the same component of G_1 if and only if $x_i^{(2)}$ and $x_j^{(2)}$ are in the same component of G_2,

(iii) for every nonnegative $s\le p$ and all $i_1,\dots,i_s\in\{1,\dots,p\}$ such that $i_1<\dots<i_s$, $(G_1,x^{(1)})$ and $(G_2,x^{(2)})$ have the same set of induced s-boundaried subgraphs in $\mathcal{F}_{\rm b}$ with respect to $(x_{i_1}^{(1)},\dots,x_{i_s}^{(1)})$ and $(x_{i_1}^{(2)},\dots,x_{i_s}^{(2)})$ respectively.

Informally, condition (i) states that the subgraphs induced by the boundaries are identical, (ii) ensures that the components of G_1 and G_2 are attached to the boundary in the same way and this allows to control connectivity, and (iii) means that G_1 and G_2 behave in the same way with respect to $\mathcal{F}_{\mathrm{b}}$.

It is straightforward to verify that the introduced relation is indeed an equivalence relation on the set of properly p-boundaried graphs. The following property of the equivalence with respect to $\mathcal{F}_{\mathrm{b}}$ is crucial for our algorithm. Roughly speaking, Lemma 4 says that we can pick any boundaried graph from the same equivalence class to represent the part of a solution that is outside (G, p).

Lemma 4. *Let (G, x), $(H_1, y^{(1)})$ and $(H_2, y^{(2)})$ be boundary-compatible p-boundaried graphs, $x = (x_1, \ldots, x_p)$ and $y^{(h)} = (y_1^{(h)}, \ldots, y_p^{(h)})$ for $h \in \{1, 2\}$. If $(H_1, y^{(1)}) \equiv_{\mathcal{F}_{\mathrm{b}}} (H_2, y^{(2)})$, then $(G, x) \oplus_{\mathrm{b}} (H_1, y^{(1)})$ is $\mathcal{F}$-free if and only if $(G, x) \oplus_{\mathrm{b}} (H_2, y^{(2)})$ is $\mathcal{F}$-free.*

Proof. By symmetry, it is sufficient to show that if $G \oplus_{\mathrm{b}} H_1$ is not $\mathcal{F}$-free, then the same holds for $G \oplus_{\mathrm{b}} H_2$. Suppose that F is an induced subgraph of $G \oplus_{\mathrm{b}} H_1$ isomorphic to a graph of $\mathcal{F}$. If $V(F) \subseteq V(G)$, then the claim is trivial. Suppose that this is not the case and $V(F) \cap V(H_1) \neq \emptyset$. Recall that $G \oplus_{\mathrm{b}} H_1$ is obtained by identifying each x_i and $y_i^{(1)}$. Denote the identified vertices by $y_1^{(1)}, \ldots, y_p^{(1)}$. Let $F_1 = F[V(F) \cap V(H_1)]$ and $F' = F[V(F) \cap V(G)]$; note that F' could be empty. Let $\{y_{i_1}^{(1)}, \ldots, y_{i_s}^{(1)}\} = V(F) \cap \{y_1^{(1)}, \ldots, y_p^{(1)}\}$ for $1 \leq i_1 < \ldots < i_s \leq p$; note that this set could be empty. Clearly, $(F_1, (y_{i_1}^{(1)}, \ldots, y_{i_s}^{(1)}))$ is an s-boundaried subgraph of H_1 with respect to $(y_{i_1}^{(1)}, \ldots, y_{i_s}^{(1)})$. Observe that $\mathcal{F}_{\mathrm{b}}$ contains an s-boundaried graph isomorphic to $(F_1, (y_{i_1}^{(1)}, \ldots, y_{i_s}^{(1)}))$. Because $(H_1, y^{(1)}) \equiv_{\mathcal{F}_{\mathrm{b}}} (H_2, y^{(2)})$, there is an induced s-boundaried subgraph $(F_2, (y_{i_1}^{(2)}, \ldots, y_{i_s}^{(2)}))$ of H_2 with respect to $(y_{i_1}^{(2)}, \ldots, y_{i_s}^{(2)})$ isomorphic to $(F_1, (y_{i_1}^{(1)}, \ldots, y_{i_s}^{(1)}))$. Then $F' \oplus_{\mathrm{b}} F_2$ is isomorphic to F, that is, $G \oplus_{\mathrm{b}} H_2$ contains F as an induced subgraph. $\square$

We can check the equivalence of two boundaried graphs in polynomial time.

Lemma 5. *For two properly p-boundaried graphs $(G_1, x^{(1)})$ and $(G_2, x^{(2)})$, it can be checked in time $(|V(G_1)| + |V(G_2)|)^{\mathcal{O}(1)}$ whether $(G_1, x^{(1)}) \equiv_{\mathcal{F}_{\mathrm{b}}} (G_2, x^{(2)})$ in such a way that the constant hidden in the $\mathcal{O}$-notation depends on $\mathcal{F}$ only.*

Proof. Let $(G_1, x^{(1)})$ and $(G_2, x^{(2)})$, where $x^{(h)} = (x_1^{(h)}, \ldots, x_p^{(h)})$ for $h \in \{1, 2\}$, be two boundaried graphs. Clearly, conditions (i) and (ii) of the definition of the equivalence with respect to $\mathcal{F}$ can be checked in polynomial time. To verify (iii), let $a = |\mathcal{F}_{\mathrm{b}}|$, b be the maximum size of the boundary of graphs in $\mathcal{F}_{\mathrm{b}}$ and let c be the maximum number of vertices of a graph in $\mathcal{F}_{\mathrm{b}}$. Clearly, the values of a, b and c depend on $\mathcal{F}$ only. For each s-tuple of indices $(i_1, \ldots, i_s)$ where $s \leq b$, we check whether an s-boundaried graph $H \in \mathcal{F}_{\mathrm{b}}$ is an s-boundaried induced subgraph of G_1 and G_2 with respect to $(x_{i_1}^{(1)}, \ldots, x_{i_s}^{(1)})$ and $(x_{i_1}^{(2)}, \ldots, x_{i_s}^{(2)})$ respectively. Since (1) there are at most bp^b s-tuples of indices $(i_1, \ldots, i_s)$, (2) at most a

graphs in $\mathcal{F}_{\mathrm{b}}$, and (3) G_h has at most $c|V(G_h)|^c$ induced subgraphs with at most c vertices for $h \in \{1,2\}$, we have that (iii) can be checked in polynomial time. $\square$

Now we show that the number of equivalence classes is bounded and we can select representatives of bounded size.

Definition 9. *For a nonnegative integer p, $\mathcal{G}_p$ is the set of properly p-boundaried graphs obtained by choosing a graph with minimum number of vertices in each equivalence class of properly p-boundaried graphs.*

We show that the size of $\mathcal{G}_p$ and the size of each graph in the set $\mathcal{G}_p$ is upper bounded by functions of p, and this set can be constructed in time that depends only on p assuming that $\mathcal{F}_{\mathrm{b}}$ is fixed. We need the following observation made by Fomin et al. [9].

Lemma 6 ([9])**.** *Let G be a connected graph and $S \subseteq V(G)$. Let F be an inclusion minimal connected induced subgraph of G such that $S \subseteq V(F)$ and let $X = \{v \in V(F) \mid d_F(v) \geq 3\} \cup S$. Then $|X| \leq 4|S| - 6$.*

Lemma 7. *For every positive integer p, $|\mathcal{G}_p| = 2^{\mathcal{O}(p^2)}$, and for every $H \in \mathcal{G}_p$, $|V(H)| = p^{\mathcal{O}(1)}$, where the constants hidden in the $\mathcal{O}$-notations depend on $\mathcal{F}$ only. Moreover, for every p-boundaried graph G, the number of p-boundaried graphs in $\mathcal{G}_p$ that are boundary-compatible with G is $2^{\mathcal{O}(p \log p)}$.*

Proof. Let $a = |\mathcal{F}_{\mathrm{b}}|$, b be the maximum size of the boundary of graphs in $\mathcal{F}_{\mathrm{b}}$ and let c be the maximum number of vertices of a graph in $\mathcal{F}_{\mathrm{b}}$. Clearly, the values of a, b, and c depend on $\mathcal{F}$ only. We fix some arbitrary boundary $x = (x_1, \ldots, x_p)$.

There are $2^{\binom{p}{2}}$ possibilities to select a set of edges with both end-vertices in $\{x_1, \ldots, x_p\}$. The number of possible partitions of the boundary into components is the Bell number $B_p = 2^{\mathcal{O}(p \log p)}$. The number of s-subtuples of $(x_1, \ldots, x_p)$ that could be boundaries of the copies of s-boundaried induced subgraphs $H \in \mathcal{F}_{\mathrm{b}}$ is at most bp^b. Consequently, the number of distinct equivalence classes is at most $2^{\binom{p}{2}} B_p b p^b 2^a$, that is, $|\mathcal{G}_p| \leq 2^{\binom{p}{2}} B_p b p^b 2^a = 2^{\mathcal{O}(p^2)}$.

Let G be a p-boundaried graph in one of the classes with minimum number of vertices. Observe that G contains at most $bp^b 2^a$ pairwise nonisomorphic boundaried induced subgraphs $H \in \mathcal{F}_{\mathrm{b}}$ with respect to pairwise distinct s-subtuples of $(x_1, \ldots, x_p)$. For each s-subtuple of $(x_1, \ldots, x_p)$, we consider all pairwise nonisomorphic boundaried induced subgraphs $H \in \mathcal{F}_{\mathrm{b}}$ that are in G with respect to the subtuple and mark the vertices of G that are in these subgraphs. Notice that for each s-subtuple, G can contain several copies of the same H as an induced subgraph with respect to this s-tuple. In this case, we pick one of these copies and mark it. Let S be the set of vertices of G that belong to these marked subgraphs or to the boundary x. We have that $|S| \leq bp^b 2^a c + p$. Let $X = \{v \in V(G) \mid d_G(v) \geq 3\} \cup S$. Since G is a p-boundaried graph with minimum number of vertices in the considered class, each component Q of G is an inclusion minimal connected induced subgraph of G containing the vertices of $V(Q) \cap S$. By applying Lemma 6 to each component of G, we obtain that $|X| \leq 4|S| - 6$.

By the minimality of G, every vertex of degree one is in S. Hence, $Y = V(G) \setminus X$ contains only vertices of degree two and, therefore, $G[Y]$ is a union of disjoint paths. Observe that by the minimality of G, each vertex of Y is a cut vertex of the component of G containing it. It implies that $G[Y]$ contains at most $|X|-1$ paths. Suppose that $G[Y]$ contains a path P with at least $2c+2$ vertices. Let G' be the graph obtained from G by the contraction of one edge of P. We claim that G and G' are equivalent with respect to $\mathcal{F}_{\mathrm{b}}$. Since the end-vertices of the contracted edges are not the vertices of the boundary, conditions (i) and (ii) of the definition of the equivalence are fulfilled. Therefore, it is sufficient to verify (iii). Let $i_1, \dots, i_s \in \{1, \dots, p\}$ and $1 \leq i_1 < \dots < i_s \leq p$. Suppose that G contains an s-boundaried induced subgraph $H \in \mathcal{F}_{\mathrm{b}}$ with respect to the s-tuple $(x_{i_1}, \dots, x_{i_s})$. Then at least two adjacent vertices of P are not included in the copy of H in G. It implies that H is an induced subgraph of G' with respect to $(x_{i_1}, \dots, x_{i_s})$. Suppose that G' contains an s-boundaried induced subgraph $H \in \mathcal{F}_{\mathrm{b}}$ with respect to the s-tuple $(x_{i_1}, \dots, x_{i_s})$. Then at least one vertex of P is not included in the copy of H in G'. Then H is an induced subgraph of G' with respect to $(x_{i_1}, \dots, x_{i_s})$. But the equivalence of G and G' contradicts the minimality of G. We conclude that each path in $G[Y]$ contains at most $2c + 1$ vertices. Then the total number of vertices of G is at most $|X| + (|X| - 1)(2c + 1) = p^{\mathcal{O}(1)}$.

To see that for any p-boundaried graph G, the number of graphs in $\mathcal{G}_p$ that are boundary-compatible with G is $2^{\mathcal{O}(p \log p)}$, notice that if $(H, (x_1, \dots, x_p))$ is in $\mathcal{G}_p$ and boundary-compatible with G, then the adjacency between the vertices of the boundary is defined by G. Then the number of s-subtuples of $(x_1, \dots, x_p)$ that could be boundaries of the copies of s-boundaried induced subgraphs from $\mathcal{F}_{\mathrm{b}}$ is at most bp^b and for each s-tuple we can have at most 2^a s-boundaried induced subgraphs from $\mathcal{F}_{\mathrm{b}}$. Taking into account that there are $2^{\mathcal{O}(p \log p)}$ possibilities for the vertices of the boundary to be partitioned according to their inclusions in the components, we obtain the claim. $\square$

We use the boundaried graphs of $\mathcal{G}_p$ to represent the parts of solutions that are outside the considered boundaried graphs (G, x). We are using similar gadget graphs to represent some parts of a (potential) solution in the reduction rules of the main step of our algorithm. Because of the similarity, we introduce the set of these boundaried graphs and give its properties here.

Definition 10. *Let $\mathcal{C}_p$ be the class of p-boundaried graphs such that a p-boundaried graph $(G, (x_1, \dots, x_p)) \in \mathcal{C}_p$ if and only if it holds that for every component H of $G - \{x_1, \dots, x_p\}$, $N_G(V(H)) = \{x_1, \dots, x_p\}$. We consider the equivalence relation $\equiv_{\mathcal{F}_{\mathrm{b}}}$ on $\mathcal{C}_p$ and define $\mathcal{G}'_p$ as follows. In each equivalence class, we select a graph $(G, (x_1, \dots, x_p)) \in \mathcal{C}$ such that (i) the number of components of $G - \{x_1, \dots, x_p\}$ is minimum and (ii) the number of vertices of G is minimum subject to (i), and then include it in $\mathcal{G}'_p$.*

The important property of graphs (G, x) of $\mathcal{C}_p$ and $\mathcal{G}'_p$ that we exploit in our algorithm is that each component of $G - x$ has "full boundary", that is, its neighborhood is x.

Lemma 8. *For every positive integer p, $|\mathcal{G}'_p| = 2^{\mathcal{O}(p^2)}$, and for each $H \in \mathcal{G}'_p$, $|V(H)| = p^{\mathcal{O}(1)}$, and the constants hidden in the $\mathcal{O}$-notations depend on $\mathcal{F}$ only. Moreover, for any p-boundaried graph G, the number of p-boundaried graphs in $\mathcal{G}'_p$ that are boundary-compatible with G is $p^{\mathcal{O}(1)}$.*

Proof. Let $a = |\mathcal{F}_{\rm b}|$, b be the maximum size of the boundary of graphs in $\mathcal{F}_{\rm b}$ and let c be the maximum number of vertices of a graph in $\mathcal{F}_{\rm b}$. Clearly, the values of a, b and c depend on $\mathcal{F}$ only. Assume that the boundary $x = (x_1, \ldots, x_p)$ is fixed.

There are $2^{\binom{p}{2}}$ possibilities to select a set of edges with both end-vertices in $\{x_1, \ldots, x_p\}$. The number of s-subtuples of $(x_1, \ldots, x_p)$ that could be boundaries of the copies of s-boundaried induced subgraphs $H \in \mathcal{F}_{\rm b}$ is at most bp^b. Consequently, the number of distinct equivalence classes of $\mathcal{C}_p$ is at most $2^{\binom{p}{2}} bp^b 2^a$, that is, $|\mathcal{G}'_p| \leq 2^{\binom{p}{2}} bp^b 2^a = 2^{\mathcal{O}(p^2)}$.

Let (G, x) be a p-boundaried graph in one of the classes such that the number of components of $G - \{x_1, \ldots, x_p\}$ is minimum and the number of vertices of G is minimum subject to the first condition. Let $Q_1, \ldots, Q_r$ be the components of $G - \{x_1, \ldots, x_p\}$. For $i \in \{1, \ldots, r\}$, let $Q'_i = G[V(Q_i) \cup \{x_1, \ldots, x_p\}]$.

Let $i \in \{1, \ldots, r\}$. We show that Q_i has $p^{\mathcal{O}(1)}$ vertices. Similarly to the proof of Lemma 7, observe that Q'_i contains at most $bp^b 2^a$ pairwise nonisomorphic boundaried induced subgraphs $H \in \mathcal{F}_{\rm b}$ with respect to pairwise distinct s-subtuples of $(x_1, \ldots, x_p)$. For each s-subtuple of $(x_1, \ldots, x_p)$, we consider all pairwise nonisomorphic boundaried induced subgraphs $H \in \mathcal{F}_{\rm b}$ that are in G with respect to the subtuple and mark the vertices of G that are in these subgraphs. Notice that for each s-subtuple, Q'_i can contain several copies of the same H as an induced subgraph with respect to this s-tuple. In this case, we pick one of these copies and mark it. Let S be the set of vertices of S that belong to these marked subgraphs or to the boundary x. We have that $|S| \leq bp^b 2^a c + p$. Let $X = \{v \in V(Q_i) \mid d_{Q'_i}(v) \geq 3\} \cup S$. By applying Lemma 6 to the graph Q''_i obtained from Q'_i by the deletion of the edges with both end-vertices in the boundary, we conclude that $|X| \leq 4|S| - 6$. To see it, note that since Q''_i has no edge with both end-vertices in the boundary, Q''_i is an inclusion minimal connected subgraph of Q'_i containing the vertices of X. Then, by the same arguments as in the proof of Lemma 7, we obtain that Q''_i and, therefore, Q'_i has at most $|X| + (|X| - 1)(2c + 1) = p^{\mathcal{O}(1)}$ vertices. Since $V(Q_i) \subseteq V(Q'_i)$, we have that Q_i has $p^{\mathcal{O}(1)}$ vertices.

We proved that each component of $G - \{x_1, \ldots, x_p\}$ has $p^{\mathcal{O}(1)}$ vertices. Now we show that the number of components is bounded.

Suppose that there are $c + 1$ pairwise distinct but equivalent $(Q'_{j_0}, x), \ldots, (Q'_{j_c}, x)$ for $j_0, \ldots, j_c \in \{1, \ldots, r\}$. Assume that G contains an s-boundaried induced subgraph $H \in \mathcal{F}_{\rm b}$ with respect to an s-tuple $(x_{i_1}, \ldots, x_{i_s})$ for some $i_1, \ldots, i_s \in \{1, \ldots, p\}$ with $1 \leq i_1 < \ldots < i_s \leq p$. Since $|V(H)| \leq c$, there is $h \in \{0, \ldots, c\}$ such that $V(H) \cap V(Q_{j_h}) = \emptyset$. Because

$(Q'_{j_0}, x), \ldots, (Q'_{j_c}, x)$ are equivalent, we obtain that H is an s-boundaried induced subgraph of $G - V(Q_{j_0})$, contradicting the minimality condition of the choice of G. Therefore, there are at most c pairwise equivalent boundaried graphs in $\{Q'_1, \ldots, Q'_r\}$.

We claim that the number of pairwise nonequivalent graphs in $\{Q'_1, \ldots, Q'_r\}$ is $p^{\mathcal{O}(1)}$. Notice that the adjacency between the boundary vertices is defined by G. Then the number of s-subtuples of $(x_1, \ldots, x_p)$ that could be boundaries of the copies of s-boundaried induced subgraphs from $\mathcal{F}_{\mathrm{b}}$ is at most bp^b and, for each s-tuple, we can have at most 2^a many s-boundaried induced subgraphs from $\mathcal{F}_{\mathrm{b}}$. Then the claim follows.

We conclude that $r = cp^{\mathcal{O}(1)}$. Since $|V(Q_i)| = p^{\mathcal{O}(1)}$ for each $i \in \{1, \ldots, r\}$, $|V(G)| = p^{\mathcal{O}(1)}$.

To see that for any p-boundaried graph G, the number of graphs in $\mathcal{G}'_p$ that are boundary-compatible with G is $p^{\mathcal{O}(1)}$, notice that if $(H, (x_1, \ldots, x_p))$ is in $\mathcal{G}_p$ and boundary-compatible with G, then the adjacency between the vertices of the boundary is defined by G. Then the number of s-tuples of vertices of $\{x_1, \ldots, x_p\}$ that could be boundaries of the copies of s-boundaried induced subgraphs from $\mathcal{F}_{\mathrm{b}}$ is at most bp^b and for each s-tuple we can have at most 2^a s-boundaried induced subgraphs from $\mathcal{F}_{\mathrm{b}}$. $\square$

Lemmas 5, 7 and 8 immediately imply that $\mathcal{G}_p$ and $\mathcal{G}'_p$ can be constructed by brute force.

Lemma 9. *The sets $\mathcal{G}_p$ and $\mathcal{G}'_p$ can be constructed in time $2^{p^{\mathcal{O}(1)}}$.*

Our next aim is to formally define the variant of Connected Secluded $\mathcal{F}$-Free Subgraph that is tailored for the recursion. Suppose that (G, x) is a p-boundaried graph that is obtained at some step of our recursion. As G is constructed by partitioning the original input graph, a (potential) solution of Connected Secluded $\mathcal{F}$-Free Subgraph is split between G and the outside part via the boundary. Since the outside parts of solutions are unknown, we represent them by graphs from $\mathcal{G}_p$. This means that we have to solve our problem for each graph of $\mathcal{G}_p$. We get a solution (if it exists) for every extension via the boundary. In this way, we obtain a list of solutions. Hence, we need a variant of Connected Secluded $\mathcal{F}$-Free Subgraph whose task is to construct such a list.

We define the problem in two steps. First, we introduce the following auxiliary problem called w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph that is defined for a given positive integer w. For given properties, we say that a weighted graph H is *w-maximum* (with respect to the properties) if the weight of H is either maximum among all graphs with the required properties or the weight of H is at least w.

w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph

Input: A graph G, sets $I, O, B \subseteq V(G)$ such that $I \cap O = \emptyset$ and $I \cap B = \emptyset$, a weight function $\omega\colon V(G) \to \mathbb{Z}_{\geq 0}$ and a non-negative integer t.

Task: Output a w-maximum t-secluded $\mathcal{F}$-free induced connected subgraph H of G such that $I \subseteq V(H)$, $O \subseteq V(G) \setminus V(H)$ and $N_G(V(H)) \subseteq B$ and output $\emptyset$ if such a subgraph does not exist.

Note that the input of the problem contains annotations of vertices, because in the process of the recursion, we make decisions for some vertices. Respectively, the set I is the set of *inside* vertices that should be in every solution, the vertices of O are *outside* vertices that are not included in any solution, and the neighborhood of H should be in the set of *border* vertices B. Initially, $I = \emptyset$, $O = \emptyset$ and B is the set of vertices of the input graph. Notice also that w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph is an optimization problem and a *solution* is either an induced subgraph H of maximum weight, or of weight at least w, or $\emptyset$. The value of the parameter w is inherited from the original considered instance of Connected Secluded $\mathcal{F}$-Free Subgraph. For technical reasons, it is convenient to have the parameter instead of solving a maximization problem, as it allows to simplify some reduction rules. The intuition behind using the parameter is following. If we get a solution of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph of weight at least w, then we already have a solution that we need and any extension is unnecessary. Otherwise, if the weight w' of a solution of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph is less than w, then we need an extension that contains additional vertices of weight at least $w - w'$.

Next, we use w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph to introduce the problem for boundaried graphs that we actually solve. For this, we need additional definitions.

Definition 11. *Let (G, I, O, B, ω, t) be an instance of w-*Maximum Connected Secluded $\mathcal{F}$-Free Subgraph *and let $T \subseteq V(G)$ be a set of* boundary terminals. *We say that an instance $(G', I', O', B', \omega', t')$ is obtained by a* boundary complementation *if there is a partition (X, Y, Z) of T (some sets could be empty), where $X = \{x_1, \ldots, x_p\}$, such that $Y = \emptyset$ if $X = \emptyset$, $I \cap T \subseteq X$, $O \cap T \subseteq Y \cup Z$ and $Y \subseteq B$, and there is a p-boundaried graph $(H, y) \in \mathcal{G}_p$ such that (H, y) and $(G, (x_1, \ldots, x_p))$ are boundary-compatible, and the following holds:*

(i) G' is obtained from $(G, (x_1, \ldots, x_p)) \oplus_{\mathrm{b}} (H, y)$ by adding edges joining every vertex of $V(H)$ with every vertex of Y,

(ii) $I' = I \cup V(H)$,

(iii) $O' = O \cup Y \cup Z$,

(iv) $B' = B \setminus X$,

(v) $\omega'(v) = \omega(v)$ *for* $v \in V(G)$ *and* $\omega'(v) = 0$ *for* $v \in V(H) \setminus X$,

(vi) $t' \leq t$.

We also say that (G', I', O', B', w', t') *is a* boundary complementation of (G, I, O, B, ω, t) with respect to $(X = \{x_1, \dots, x_p\}, Y, Z, H)$.

In the definition, we keep the notation $X = \{x_1, \dots, x_p\}$ for the set of vertices obtained by the identification in the boundary sum.

We say that $(X = \{x_1, \dots, x_p\}, Y, Z, H)$ is *feasible* for (G, I, O, B, ω, t) if it holds that $Y = \emptyset$ if $X = \emptyset$, $I \cap T \subseteq X$, $O \cap T \subseteq Y \cup Z$ and $Y \subseteq B$, and the p-boundaried graph $H \in \mathcal{G}_p$ and $(G, (x_1, \dots, x_p))$ are boundary-compatible.

The construction of a boundary complementation is shown in Fig. 2. The set T forms the boundary of G. Recall that we represent the parts of solutions of Connected Secluded $\mathcal{F}$-Free Subgraph that are outside G by graphs of $\mathcal{G}_p$. Respectively, we attach such a graph to a part of the boundary (the set X) that is included in a solution assuming that the weights of the added vertices is zero. The remaining vertices of T, that is, the vertices of $Y \cup Z$ are not included in any solution. We also have to take into account that the vertices of a solution that are outside G can have neighbors in T, and we use Y to encode this. Note that if $X = \emptyset$, that is, the part of a solution outside G is empty, then Y should be empty as well.

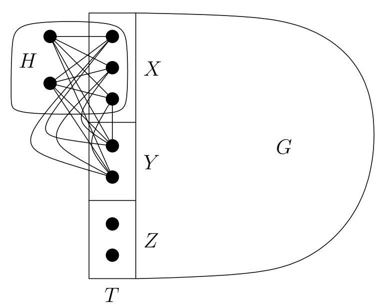

Figure 2: Construction of a boundary complementation

Now we define the problem that is actually solved by our algorithm.

BOUNDARIED w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH

Input: A graph G, sets $I, O, B \subseteq V(G)$ such that $I \cap O = \emptyset$ and $I \cap B = \emptyset$, a weight function $\omega\colon V(G) \to \mathbb{Z}_{\geq 0}$, a nonnegative integer t, and a set $T \subseteq V(G)$ of boundary terminals of size at most $2t$.

Task: For each instance (G', I', O', B', w', t') of w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH that can be obtained from (G, I, O, B, w, t) by a boundary complementation, output

- a nonempty solution, if it has weight at least w, and $\emptyset$, otherwise, for the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$,
- a solution for all other boundary complementations.

Note that BOUNDARIED w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH is neither a decision nor an optimization problem as its task is to list connected subgraphs that are solutions of w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH for all possible boundary complementations. In some cases we have no solution of w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH and this is encoded by the inclusion of $\emptyset$ for the corresponding boundary complementation. Slightly abusing notation, we assume that $\emptyset$ is also a graph. Then we have that a solution of an instance of BOUNDARIED w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH is a list of connected graphs. Observe also that the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$ is a special case as we are looking for a solution that contains no vertex outside G. Respectively, if we fail to find a solution that is a solution for the original instance of CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH, we have no solution and output $\emptyset$ to indicate this. The crucial properties of the list of solutions are the following:

(i) the part of every solution for the original input instance of CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH that is outside G can be extended by a solution from the list,

(ii) the size of the list is upper-bounded by a function of the parameter t.

This allows us to use the lists obtained by solving instances of BOUNDARIED w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH to solve CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH for the input instance.

Given an instance of BOUNDARIED w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH and its solution, we reduce the size of G. To be able to perform this reduction, we have to define the safeness for reduction rules. For this, we need the following definition.

Definition 12. *Two instances* $(G_1, I_1, O_1, B_1, \omega_1, t, T)$ *and* $(G_2, I_2, O_2, B_2, \omega_2, t, T)$ *of* Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph *(note that the instances agree on* t *and* T*) are said to be* equivalent *if*

(i) $T \cap I_1 = T \cap I_2$*,* $T \cap O_1 = T \cap O_2$ *and* $T \cap B_1 = T \cap B_2$*,*

(ii) *for the boundary complementations* $(G_1', I_1', O_1', B_1', \omega_1', t')$ *and* $(G_2', I_2', O_2', B_2', \omega_2', t')$ *of the instances* $(G_1, I_1, O_1, B_1, \omega_1, t)$ *and* $(G_2, I_2, O_2, B_2, \omega_2, t)$ *respectively of* w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph *with respect to every feasible* $(X = \{x_1, \ldots, x_p\}, Y, Z, H)$ *and* $t' \leq t$*, it holds that*

   (a) *if* $(G_1', I_1', O_1', B_1', \omega_1', t')$ *has a nonempty solution* R_1*, then* $(G_2', I_2', O_2', B_2', \omega_2', t')$ *has a nonempty solution* R_2 *with* $w_2'(V(R_2)) \geq \min\{\omega_1'(V(R_1)), w\}$ *and, vice versa,*

   (b) *if* $(G_2', I_2', O_2', B_2', \omega_2', t')$ *has a nonempty solution* R_2*, then* $(G_1', I_1', O_1', B_1', \omega_1', t')$ *has a nonempty solution* R_1 *with* $\omega_1'(V(R_1)) \geq \min\{\omega_2'(V(R_2)), w\}$*.*

We will say that a reduction rule is *safe* if it produces an equivalent instance. Note that if $(G_1, I_1, O_1, B_1, \omega_1, t, T)$ and $(G_2, I_2, O_2, B_2, \omega_2, t, T)$ are equivalent, then a solution of the first problem is not necessarily a solution of the second. Nevertheless, we know that for each boundary complementation, we can replace the solution for each instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising for $(G_1, I_1, O_1, B_1, \omega_1, t, T)$ by the solution for the corresponding instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for $(G_2, I_2, O_2, B_2, \omega_2, t, T)$ and vice versa. This allows us to not distinguish equivalent instances of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph and their solutions.

We conclude this subsection by giving an informal short overview of our algorithm solving Connected Secluded $\mathcal{F}$-Free Subgraph.

Given an instance (G, ω, t, w), we construct the initial instance of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for w by setting $T = \emptyset$, $I = \emptyset$, $O = \emptyset$ and $B = V(G)$ and run our algorithm for this instance. Clearly, (G, ω, t, w) is a yes-instance of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph if and only if a solution for the corresponding instance of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph contains a connected subgraph R with $\omega(V(R)) \geq w$.

To explain the idea of the recursion, we assume that we are given an instance $(G, I, O, B, \omega, t, T)$ of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph. We choose an appropriate value $q \in 2^{2^{\mathcal{O}(t \log t)}}$ (the choice of q is tailored to ensure that we obtain an FPT algorithm) and apply Lemma 3 for $k = t$. If the algorithm from Lemma 3 reports that G is $((2q+1)q \cdot 2^t, t)$-unbreakable, then we solve Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for the considered instance directly using the fact that G

is highly connected. The intuition behind our algorithm for this case is that because of unbreakability, every instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph that arises in Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph has a solution that is either small in the sense that it has size bounded by some function of t or very big, that is, the size of the part of the graph that is not included in the solution is bounded by some function of the parameter. To find small solutions we use Theorem 2 and to deal with large solutions we use the *important separator* technique introduced by Marx [18].

Assume from now on that there is a separation (U, W) of G of order at most t such that $|U \setminus W| > q$ and $|W \setminus U| > q$. By definition, $|T| \leq 2t$. This implies that either $|(U \setminus W) \cap T| \leq t$ or $|(W \setminus U) \cap T| \leq t$. Let us assume without loss of generality that $|(W \setminus U) \cap T| \leq t$. Let $\tilde{G} = G[W]$ and $\tilde{T} = (U \cap W) \cup (T \cap B)$. Note that $|\tilde{T}| \leq 2t$. Clearly, $|V(\tilde{G})| < |V(G)| - q$. We apply our algorithm for Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph recursively for $\tilde{G}$ with the set of boundary terminals $\tilde{T}$, where the weights and the annotations of the vertices are inherited from ω and the annotations of the vertices of G. The algorithm produces the list $\mathcal{R}$ of solutions of the instances of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph. The crucial property of $\mathcal{R}$ is that its elements represent partial solutions of $(G, I, O, B, \omega, t, T)$ inside $G[W]$. As we noted above, the size of $\mathcal{R}$ is bounded by a function of t. A solution from $\mathcal{R}$ can have an arbitrary size, but the size of its neighborhood is upper bounded by t. Let S be the union of all the neighborhoods. We obtain that $|S| \leq t|\mathcal{R}|$. We use this property to modify $(G, I, O, B, \omega, t, T)$. First, we can replace B by $B^* = B \cap (U \cup S)$. Let $S' = T' \cup S$. For every component Q of $\tilde{G} - S'$, we have that for every solution, either Q is completely in the solution or is completely outside it. This allows us to discard some components. The crucial reduction is that, afterwards, we can replace the set of components Q of $\tilde{G} - S'$ with the same neighborhood in S' by a single gadget of bounded size using graphs from the families $\mathcal{G}'_p$. This way, we construct an instance $(G^*, I^*, O^*, B^*, \omega^*, t, T)$ of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph that is equivalent to $(G, I, O, B, \omega, t, T)$ such that the size of $G^* - U$ is bounded by a function of t. This means that the size of G^* is strictly less than the size of G. Then we recurse on $(G^*, I^*, O^*, B^*, \omega^*, t, T^*)$.

4.2. High connectivity phase

In this section we solve Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for (q, t)-unbreakable graphs. For this purpose, we use *important separators* defined by Marx [18]. Essentially, we follow the terminology given by Cygan et al. [6]. Recall that for $X, Y \subseteq V(G)$, a set $S \subseteq V(G)$ is an (X, Y)-separator if $G - S$ has no path joining a vertex of $X \setminus S$ with a vertex of $Y \setminus S$. An (X, Y)-separator S is *minimal* if no proper subset of S is an (X, Y)-separator. For $X \subseteq V(G)$ and $v \in V(G)$, it is said that v is *reachable* from X if there is an (x, v)-path in G with $x \in X$. A minimal (X, Y)-separator S can be characterized by the set of vertices reachable from $X \setminus S$ in $G - S$.

Lemma 10 ([6])**.** *If S is a minimal (X, Y)-separator in G, then $S = N_G(Z)$ where Z is the set of vertices reachable from $X \setminus S$ in $G - S$.*

Definition 13. *Let $X, Y \subseteq V(G)$ for a graph G. Let $S \subseteq V(G)$ be an (X, Y)-separator and let Z be the set of vertices reachable from $X \setminus S$ in $G - S$. It is said that S is an* important (X, Y)-separator *if*

(i) S is minimal, and

(ii) there is no (X, Y)-separator $S' \subseteq V(G)$ with $|S'| \leq |S|$ such that $Z \subset Z'$ where Z' is the set of vertices reachable from $X \setminus S'$ in $G - S'$.

Lemma 11 ([6])**.** *Let $X, Y \subseteq V(G)$ for a graph G, let t be a nonnegative integer and let $\mathcal{S}_t$ be the set of all important (X, Y)-separators of size at most t. Then $|\mathcal{S}_t| \leq 4^t$ and $\mathcal{S}_t$ can be constructed in time $\mathcal{O}(|\mathcal{S}_t| \cdot t^2 \cdot (n + m))$.*

The following lemma shows that for every graph R in a solution of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph, it holds that either $|V(R)|$ or $|V(G) \setminus V(R)|$ has bounded size.

Lemma 12. *Let $(G, I, O, B, \omega, t, T)$ be an instance of* Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph *where G is a (q, t)-unbreakable graph for a positive integer q. Then for each nonempty graph R in a solution of* Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph*, either $|V(R) \cap V(G)| \leq q$ or $|V(G) \setminus V(R)| \leq q + t$.*

Proof. Let R be a nonempty graph in a solution of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for an instance $(G', I', O', B', \omega', t')$ of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph. Assume that G' is obtained from $(G, (x_1, \ldots, x_p)) \oplus_{\mathrm{b}} (H, y)$ for $H \in \mathcal{G}_p$, as explained in Definition 11 (i). Let $U = N_{G'}[V(R) \cap V(G)]$ and $W = V(G) \setminus V(R)$. Clearly, (U, W) is a separation of G of order at most t, because $U \cap W = N_{G'}(V(R))$ and $|N_{G'}(V(R))| \leq t$ since R is a t-secluded subgraph of G'. Since G is (q, t)-unbreakable, either $|U \setminus W| \leq q$ or $|W \setminus U| \leq q$. If $|U \setminus W| \leq q$, then $|V(R) \cap V(G)| \leq |U \setminus W| \leq q$. If $|W \setminus U| \leq q$, then $|V(G) \setminus V(R)| \leq q + t$. $\square$

Now we can prove the following crucial lemma.

Lemma 13. Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph *for (q, t)-unbreakable graphs can be solved in time $2^{\mathcal{O}(q + t \log(q+t))} \cdot n^{\mathcal{O}(1)}$ if the sets $\mathcal{G}_p$ for all $p \leq 2t$ are given.*

Proof. Assume that the sets $\mathcal{G}_p$ for all $p \leq 2t$ are given. We consider all possible instances $(G', I', O', B', \omega', t')$ of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained from the input instance $(G, I, O, B, \omega, t, T)$ of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph by boundary complementation. To construct each instance, we consider all at most 3^{2t} partitions (X, Y, Z) of T, where $X = \{x_1, \ldots, x_p\}$, such that $Y = \emptyset$ if $X = \emptyset$, $I \cap T \subseteq X$, $O \cap T \subseteq Y \cup Z$ and $Y \subseteq B$. Then we consider all

p-boundaried graphs $(H, y) \in \mathcal{G}_p$ such that $(H, y) \simeq_{\mathrm{b}} (G, (x_1, \ldots, x_p))$. By Lemma 7, there are $2^{\mathcal{O}(t \log t)}$ such boundaried graphs. Consider now a constructed instance $(G', I', O', B', \omega', t')$ and assume that G' is obtained from $(G, (x_1, \ldots, x_p)) \oplus_{\mathrm{b}} (H, y)$ for $(H, y) \in \mathcal{G}_p$ as it is explained in Definition 11 (i). We find a t-secluded $\mathcal{F}$-free induced connected subgraph R of G' of maximum weight such that $I' \subseteq V(R)$, $O' \subseteq V(G') \setminus V(R)$ and $N_{G'}(V(R)) \subseteq B'$ if such a subgraph exists. By Lemma 12, either $|V(R) \cap V(G)| \leq q$ or $|V(G) \setminus V(R)| \leq q + t$. Using this property, we separately find a t-secluded $\mathcal{F}$-free induced connected subgraph R of G' of maximum weight such that $|V(R) \cap V(G)| \leq q$ and a t-secluded $\mathcal{F}$-free induced connected subgraph R of G' of maximum weight such that $|V(G) \setminus V(R)| \leq q + t$. Then we compare the obtained graphs and output the graph of maximum weights.

Finding a small solution. First, we find a t-secluded $\mathcal{F}$-free induced connected subgraph R of G' with $|V(R) \cap V(G)| \leq q$ of maximum weight such that $I' \subseteq V(R)$, $O' \subseteq V(G') \setminus V(R)$ and $N_{G'}(V(R)) \subseteq B'$. If $|V(R) \cap V(G)| \leq q$, then $|V(R)| \leq |V(H)| + |V(R) \cap V(G)| < |V(H)| + q$. By Lemma 7, $|V(H)| = t^c$ for some constant $c \geq 1$. This implies that $|V(R)| \leq t^c + q$. To find R, we consider all $k \leq t^c + q$ and find a t-secluded $\mathcal{F}$-free induced connected subgraph R of G' of maximum weight such that $I' \subseteq V(R)$, $O' \subseteq V(G') \setminus V(R)$, $N_{G'}(V(R)) \subseteq B'$ and $|V(R)| = k$. Notice that the properties of R can be easily encoded as a property of an induced subgraph of G' with vertex colors, where the colors assigned to the vertices of G' distinguish the sets I', $O' \cap B'$, $O' \setminus B'$, $B' \setminus O'$, and $V(G') \setminus (I' \cup O' \cup B')$. Then, by Corollary 1, all these subgraphs R can be found in time $2^{\mathcal{O}(t \log(t^c + q + t))} \cdot n^{\mathcal{O}(1)}$.

Finding a large solution. Now we find a t-secluded $\mathcal{F}$-free induced connected subgraph R of G' with $|V(G) \setminus V(R)| \leq q + t$ of maximum weight such that $I' \subseteq V(R)$, $O' \subseteq V(G') \setminus V(R)$ and $N_{G'}(V(R)) \subseteq B'$.

Because $|V(G) \setminus V(R)| \leq q + t$, there is a set S such that $O' \subseteq S \subseteq V(G) \setminus V(R)$, $|S| \leq q + t$ and $G' - S$ is $\mathcal{F}$-free. We list all such sets S using the standard branching algorithm for this problem (see, e.g., [6]). The main idea of the algorithm is that if G' has an induced subgraph F isomorphic to a graph of $\mathcal{F}$, then at least one vertex of F should be in S. Initially we set $S = O'$ and set a branching parameter $h = q + t - |O'|$. If $h < 0$, we stop as there is no solution. Assume that this is not the case. We check whether $G' - S$ has an induced subgraph F isomorphic to a graph of $\mathcal{F}$. If we have no such graph, we return S. If $V(F) \subseteq I'$, then we stop. Otherwise, we branch on the vertices of F. For each $v \in V(F) \setminus I'$, we set $S := S \cup \{v\}$, set $h := h - 1$ and recurse. It is straightforward to verify the correctness of the algorithm and see that it runs in time $2^{\mathcal{O}(q+t)} \cdot n^{\mathcal{O}(1)}$, because $\mathcal{F}$ is fixed and each graph from this set has a constant size. If the algorithm fails to output any set S, then we conclude that $(G', I', O', B', \omega', t')$ has no solution R with $|V(G) \setminus V(R)| \leq q + t$ and we return $\emptyset$. From now on we assume that this is not the case.

To find a solution R of maximum weight, we consider all possible sets S, and for each S, we find a solution of maximum weight containing S. Then we choose a solution of maximum weight among all constructed solutions. Clearly, it will give us an optimum solution.

Assume that S is fixed. We set $O' := O' \cup S$. Now our task is to find a solution for the modified instance $(G', I', O', B', \omega', t')$.

If $I' = \emptyset$, we guess a vertex $u \in V(G') \setminus O'$ that is included in a solution. We set $I' := \{u\}$ and $B' := B' \setminus \{u\}$ and solve the modified instance (G', I', O', B', w', t'). Then we choose a solution of maximum weight for all guesses of u. From now on we have $I' \neq \emptyset$.

We apply a series of reduction rules for $(G', I', O', B', \omega', t')$. The aim of the rules is either to find a trivial solution or to simplify the considered instance. We show that these rules are *safe*, that is, they either correctly solve the problem or produce an instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph such that every solution for $(G', I', O', B', \omega', t')$ is a solution for the obtained instance and vice versa. Let initially $h = q + t$.

Reduction Rule 4.1. If G' is disconnected and has vertices of I' in distinct components, then return no. Otherwise, let Q be a component of G' containing I' and set $G' := Q$, $B' := B' \cap V(Q)$, $O' := O' \cap V(Q)$ and $h := h - |V(G) \setminus V(Q)|$. If $h < 0$, then return no.

It is straightforward to see that the rule is safe, because a solution is a connected graph. Notice that from now on we can assume that G' is connected.

If $O' = \emptyset$, then $S = \emptyset$ and G' is $\mathcal{F}$-free. This means that G' itself is a solution, because G' is connected and $N_{G'}(V(G')) = \emptyset$. This gives the next rule.

Reduction Rule 4.2. If $O' = \emptyset$, then return G'.

From now on we assume that $O' \neq \emptyset$. Next, we try to extend annotations I' and O'. The following rule is applied for each component Q exactly once.

Reduction Rule 4.3. For a component Q of $G' - B'$ do the following in the given order:

- if $V(Q) \cap I' \neq \emptyset$ and $V(Q) \cap O' \neq \emptyset$, then return no,
- if $V(Q) \cap I' \neq \emptyset$, then set $I' := I' \cup V(Q)$,
- if $V(Q) \cap O' \neq \emptyset$, then set $O' := O' \cup N_{G'}[V(Q)]$.

Claim 4.1. *Reduction Rule 4.3 is safe.*

Proof of Claim 4.1. Let Q be a component of $G' - B'$. Notice that for any solution R, either $V(Q) \subseteq V(R)$ or $V(Q) \cap V(R) = \emptyset$, because $N_{G'}(V(R)) \subseteq B'$. Moreover, if $V(Q) \cap V(R) = \emptyset$, then $N_{G'}[V(Q)] \cap V(R) = \emptyset$. This immediately implies safeness. $\square$

Now our aim is to find all inclusion maximal induced subgraphs R of G' such that $I' \subseteq V(R)$, $O' \cap V(R) = \emptyset$, $N_{G'}(V(R)) \subseteq B'$, $|N_{G'}(V(R))| \leq t'$ and all the vertices of R are reachable from I'. Then, by maximality, a solution is such a subgraph R that is connected and, subject to connectivity, has maximum weight.

We find all these subgraphs R using important $(N_{G'}[I'], O')$-separators. The obstacle for the application of the important separator technique of Marx [18] is the constraint that $N_{G'}(V(R)) \subseteq B'$. To overcome this obstacle, we modify G'.

Let Q be a component of $G' - B'$. By Reduction Rule 4.3, we have that exactly one of the following holds: either (i) $V(Q) \subseteq I'$ or (ii) $V(Q) \subseteq O'$ or (iii) $V(Q) \cap I' = V(Q) \cap O' = \emptyset$. To ensure that $N_{G'}(V(R)) \subseteq B'$, we have to ensure that if (iii) is fulfilled, then it holds that either $V(Q) \subseteq V(R)$ or $V(Q) \cap V(R) = \emptyset$. To do it, we construct the auxiliary graph G'' as follows. For each $v \in V(G') \setminus (I' \cup O' \cup B')$, we replace v by $t+1$ true twin vertices $v_0, \ldots, v_t$ that are adjacent to the same vertices as v in G or to the corresponding true twins obtained from the neighbors of v. For an induced subgraph R of G', we say that the induced subgraph R' of G'' is an *image* of R if R' is obtained by the same replacement of the vertices $v \in V(R) \setminus (I' \cup O' \cup B')$ by $t+1$ twins. Respectively, we say that R is a *preimage* of R'.

We show the following claim.

Claim 4.2.

- *If R is an induced subgraph of G' such that $I' \subseteq V(R)$, $O' \cap V(R) = \emptyset$, $N_{G'}(V(R)) \subseteq B'$, $|N_{G'}(V(R))| \leq t'$ and all the vertices of R are reachable from I', then its image R' is an induced subgraph of G'' such that $I' \subseteq V(R')$, $O' \cap V(R') = \emptyset$, $|N_{G''}(V(R'))| \leq t'$ and all the vertices of R' are reachable from I'.*

- *If R' is an inclusion maximal induced subgraph of G'' such that $I' \subseteq V(R')$, $O' \cap V(R') = \emptyset$, $|N_{G''}(V(R'))| \leq t'$ and all the vertices of R' are reachable from I', then R' has a preimage R and $N_{G'}(V(R)) = N_{G''}(V(R')) \subseteq B'$.*

Proof of Claim. The first part of the claim is straightforward from the definition. To prove the second part, consider an inclusion maximal induced subgraph R' of G'' such that $I' \subseteq V(R')$, $O' \cap V(R') = \emptyset$, $|N_{G''}(V(R'))| \leq t'$ and all the vertices of R' are reachable from I'. Let $v' \in N_{G''}(R')$ and assume that $v' \notin B'$. Clearly, $v' \notin I'$. Notice that $v' \notin O'$, because by Reduction Rule 4.3, we have that for any $w \in O' \setminus B'$, $N_{G'}[w] \subseteq O'$. Since $v' \notin B' \cup I' \cup O'$, $v' \in \{v_0, \ldots, v_t\}$ for $t+1$ true twins constructed for some vertex $v \in V(G')$. Because $|N_{G''}(V(R'))| \leq t'$, there is an $i \in \{0, \ldots, t\}$, such that $v_i \notin N_{G''}(R')$. As v_i and v' are twins, $v_i \in V(R')$. Let $R'' = G''[V(R') \cup \{v_0, \ldots, v_t\}]$. We obtain that $I' \subseteq V(R'')$, $O' \cap V(R'') = \emptyset$, $|N_{G''}(V(R''))| \leq t'$ and all the vertices of R'' are reachable from I', but $V(R') \subset V(R'')$, contradicting maximality. Hence, $N_{G''}(V(R')) \subseteq B'$. Then R' has a preimage R and $N_{G'}(V(R)) = N_{G''}(V(R')) \subseteq B'$. $\square$

By Claim 4.2, we conclude that to find all inclusion maximal induced subgraphs R of G' such that $I' \subseteq V(R)$, $O' \cap V(R) = \emptyset$, $N_{G'}(V(R)) \subseteq B'$, $|N_{G'}(V(R))| \leq t'$ and all the vertices of R are reachable from I', we should list inclusion maximal induced subgraphs R' of G'' such that $I' \subseteq V(R')$,

$O' \cap V(R') = \emptyset$, $|N_{G''}(V(R'))| \leq t'$ and all the vertices of R' are reachable from I', and then take preimages of the graphs R'.

To find the maximal induced subgraphs R' of G'' such that $I' \subseteq V(R')$, $O' \cap V(R') = \emptyset$, $|N_{G''}(V(R'))| \leq t'$ and all the vertices of R' are reachable from I', we use Lemma 11. In time $4^t \cdot n^{\mathcal{O}(1)}$ we construct the set $\mathcal{S}_{t'}$ of all important $(N_{G''}[I'], O')$-separators of size at most t' in G''. Then for each $S \in \mathcal{S}_{t'}$, we find R' that is the union of the components of $G'' - S$ containing the vertices of I'.

Now we have the set $\mathcal{R}$ of all inclusion maximal induced subgraphs R of G' such that $I' \subseteq V(R)$, $O' \cap V(R) = \emptyset$, $N_{G'}(V(R)) \subseteq B'$, $|N_{G'}(V(R))| \leq t'$ and all the vertices of R are reachable from I'. By Lemma 11, $|\mathcal{R}| \leq 4^t$. This set may contain disconnected graphs that could not be solutions of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph. Therefore, we delete these graphs from $\mathcal{R}$. Finally, we find a graph R of maximum weight in $\mathcal{R}$ and return it.

Since Reduction Rules 4.1–4.3 can be applied in polynomial time and G'' can be constructed in polynomial time, we have that the total running time is $2^{\mathcal{O}(t+q)} \cdot n^{\mathcal{O}(1)}$.

Now we compare the two subgraphs R that we found for the cases $|V(R) \cap V(G)| \leq q$ and $|V(G) \setminus V(R)| \leq q + t$ and output the subgraph of maximum weight or the empty set if we failed to find these subgraphs. Taking into account the time used to construct the instances $(G', I', O', B', \omega', t')$, we obtain that the total running time is $2^{\mathcal{O}(q+t\log(q+t))} \cdot n^{\mathcal{O}(1)}$. ☐

4.3. The FPT *algorithm for* Connected Secluded $\mathcal{F}$-Free Subgraph

In this section we construct an FPT algorithm for Connected Secluded $\mathcal{F}$-Free Subgraph parameterized by t. We do this by solving Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph in FPT-time for the general case.

Lemma 14. Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph *can be solved in time* $2^{2^{2^{\mathcal{O}(t\log t)}}} \cdot n^{\mathcal{O}(1)}$.

Proof. Given $\mathcal{F}$, we construct the set $\mathcal{F}_{\mathrm{b}}$. Then we use Lemma 9 to construct the sets $\mathcal{G}_p$ for all $p \in \{0, \ldots, t\}$ in time $2^{t^{\mathcal{O}(1)}}$.

By Lemma 7, there is a constant $c \geq 1$ that depends only on $\mathcal{F}$ such that for every nonnegative p and for any p-boundaried graph G, there are at most $2^{cp\log p}$ many p-boundaried graphs in $\mathcal{G}_p$ that are boundary-compatible with G and there are at most p^c many p-boundaried graphs in $\mathcal{G}'_p$ that are boundary-compatible with G. This value of c is used throughout the proof. We define

$$\begin{aligned} q =& 2^{((t+1)t3^{2t}2^{c2t\log(2t)}+2t)} \cdot 2((t+1)t3^{2t}2^{c2t\log(2t)} + 2t)^c t^c \\ &+ (t+1)t3^{2t}2^{c2t\log(2t)} + 2t. \end{aligned} \tag{1}$$

Notice that $q \in 2^{2^{\mathcal{O}(t\log t)}}$. The choice of q is defined according to the general scheme for the recursive understanding technique [5]. Informally, q should be sufficiently big to ensure that if we have a separation (U, W) of order at most t

such that $|U \setminus W| > q$ and $|W \setminus U| > q$, then we should be able to compress the input instance to call our algorithm recursively.

Consider an instance $(G, I, O, B, \omega, t, T)$ of BOUNDARIED w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH.

We use the algorithm from Lemma 3 for G. This algorithm in time $2^{2^{\mathcal{O}(t \log t)}} \cdot n^{\mathcal{O}(1)}$ either finds a separation (U, W) of G of order at most t such that $|U \setminus W| > q$ and $|W \setminus U| > q$ or correctly reports that G is $((2q+1)q \cdot 2^t, t)$-unbreakable. In the latter case we solve the problem using Lemma 13 in time $2^{2^{2^{\mathcal{O}(t \log t)}}} \cdot n^{\mathcal{O}(1)}$. Assume from now on that there is a separation (U, W) of order at most t such that $|U \setminus W| > q$ and $|W \setminus U| > q$.

Recall that $|T| \leq 2t$. Then $|T \cap (U \setminus W)| \leq t$ or $|T \cap (W \setminus U)| \leq t$. Assume without loss of generality that $|T \cap (W \setminus U)| \leq t$. Let $\tilde{G} = G[W]$, $\tilde{I} = I \cap W$, $\tilde{O} = O \cap W$, $\tilde{\omega}$ is the restriction of ω to W, and define $\tilde{T} = (T \cap W) \cup (U \cap W)$. Since $|U \cap W| \leq t$, $|\tilde{T}| \leq 2t$.

If $|W| \leq (2q+1)q \cdot 2^t$, then we solve BOUNDARIED w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH for the instance $(\tilde{G}, \tilde{I}, \tilde{O}, \tilde{B}, \tilde{\omega}, t, \tilde{T})$ by brute force in time $2^{2^{2^{\mathcal{O}(t \log t)}}}$ trying all possible subsets of W and at most $t+1$ values of $0 \leq t' \leq t$. Otherwise, we solve $(\tilde{G}, \tilde{I}, \tilde{O}, \tilde{B}, \tilde{\omega}, t, \tilde{T})$ recursively. Let $\mathcal{R}$ be the set of nonempty induced subgraphs R that are included in the obtained solution for $(\tilde{G}, \tilde{I}, \tilde{O}, \tilde{B}, \tilde{\omega}, t, \tilde{T})$.

For $R \in \mathcal{R}$, define S_R to be the set of vertices of $W \setminus V(R)$ that are adjacent to the vertices of R in the graph obtained by the boundary complementation for which R is a solution of the corresponding instance of w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH. Note that $|S_R| \leq t$. If $\mathcal{R} \neq \emptyset$, then let $S = \tilde{T} \cup (\bigcup_{R \in \mathcal{R}} S_R)$, and $S = \tilde{T}$ if $\mathcal{R} = \emptyset$. Since w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH is solved for at most $t+1$ values of $t' \leq t$, at most 3^{2t} three-partitions (X, Y, Z) of $\tilde{T}$ and at most $2^{c2t \log(2t)}$ choices of a p-boundaried graph $H \in \mathcal{F}_{\mathrm{b}}$ for $p = |X|$, we have that $|\mathcal{R}| \leq (t+1)3^{2t}2^{c2t \log(2t)}$. Taking into account that $|T'| \leq 2t$,

$$|S| \leq (t+1)t3^{2t}2^{c2t \log(2t)} + 2t, \tag{2}$$

that is, the size of S is upper bounded by a function of the parameter. This allows us to reduce the size of G.

First, we reduce the set B of the vertices that could be adjacent to solutions. Let $\hat{B} = B \cap (U \cup S)$. We prove the following claim.

Claim 4.3. *The instances* $(G, I, O, B, \omega, t, T)$ *and* $(G, I, O, \hat{B}, \omega, t, T)$ *of* BOUNDARIED w-MAXIMUM CONNECTED SECLUDED $\mathcal{F}$-FREE SUBGRAPH *are equivalent.*

Proof of Claim 4.3. Recall that by Definition 12, we have to show that

(i) $T \cap B = T \cap \hat{B}$,

(ii) for the boundary complementations $(G', I', O', B', \omega', t')$ and $(G', I', O', \hat{B}', \omega', t')$ of the instances (G, I, O, B, ω, t') and $(G, I, O, \hat{B}, \omega, t)$

respectively of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph with respect to every feasible $(X = \{x_1, \ldots, x_p\}, Y, Z, H)$ and $t' \leq t$, it holds that if $(G', I', O', B', \omega', t')$ has a nonempty solution R_1, then $(G', I', O', \hat{B}', \omega', t')$ has a nonempty solution R_2 with $\omega'(V(R_2)) \geq \min\{\omega'(V(R_1)), w\}$ and, vice versa, if $(G', I', O', \hat{B}', \omega', t')$ has a nonempty solution R_2, then $(G', I', O', B', \omega', t')$ has a nonempty solution R_1 with $\omega'(V(R_1)) \geq \min\{\omega'(V(R_2)), w\}$.

Condition (i) holds by the definition of $\hat{B}$. Because $\hat{B} \subseteq B$, we immediately obtain that if $(G', I', O', \hat{B}', \omega', t')$ has a nonempty solution R_2, then $(G', I', O', B', \omega', t')$ has a nonempty solution R_1 with $\omega'(V(R_1)) \geq \min\{\omega'(V(R_2)), w\}$. It remains to prove that for a boundary complementation $(G', I', O', B', \omega', t')$ and $(G', I', O', \hat{B}', \omega', t')$ of (G, I, O, B, ω, t') and $(G, I, O, \hat{B}, \omega, t')$ respectively of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph with respect to a feasible $(X = \{x_1, \ldots, x_p\}, Y, Z, H)$ and $t' \leq t$, it holds that if $(G', I', O', B', \omega', t')$ has a nonempty solution R_1, then $(G', I', O', \hat{B}', \omega', t')$ has a nonempty solution R_2 with $\omega'(V(R_2)) \geq \min\{\omega'(V(R_1)), w\}$.

If $V(R_1) \cap V(G) \subseteq U \setminus W$, then $N_{G'}(V(R_1)) \subseteq \hat{B}'$. Therefore, there is a solution R_2 of $(G', I', O', \hat{B}', \omega', t')$ such that $\omega'(V(R_2)) \geq \min\{\omega'(V(R_1)), w\}$. Assume that $V(R_1) \cap W \neq \emptyset$.

Recall that $\tilde{G} = G[W]$, $\tilde{I} = I \cap W$, $\tilde{O} = O \cap W$, $\tilde{\omega}$ is the restriction of ω to W, and $\tilde{T} = (T \cap W) \cup (U \cap W)$. Let $\tilde{X} = \tilde{T} \cap (V(R_1) \cap W) = \{\tilde{x}_1, \ldots, \tilde{x}_r\}$, let $\tilde{Y}$ be the set of vertices of $\tilde{T} \setminus V(R_1)$ that are adjacent to vertices of R_1 outside $W \setminus U$ and $\tilde{Z} = \tilde{T} \setminus (\tilde{X} \cup \tilde{Y})$. Let $(R_1', (\tilde{x}_1, \ldots, \tilde{x}_r))$ be the r-boundaried graph obtained from R_1 by the deletion of the vertices of $W \setminus U$ (note that the graph could be empty). We have that $\mathcal{G}_r$ contains an r-boundaried graph $\tilde{H}$ that is equivalent to $(R_1', (\tilde{x}_1, \ldots, \tilde{x}_r))$ with respect to $\mathcal{F}_{\rm b}$. Recall that we have a solution of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for $(\tilde{G}, \tilde{I}, \tilde{O}, \tilde{B}, \tilde{\omega}, t, \tilde{T})$. In particular, we have a solution $\tilde{R} \in \mathcal{R}$ for the instance $(\tilde{G}, \tilde{I}, \tilde{O}, \tilde{B}, \tilde{\omega}, \tilde{t})$ of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained by the boundary complementation with respect to $(\tilde{X} = (\tilde{x}_1, \ldots, \tilde{x}_r), \tilde{Y}, \tilde{Z}, \tilde{H})$, where $\tilde{t}$ is the number of neighbors of R_1 in W. Recall also that the neighbors of the vertices of $\tilde{R}$ are in S. Denote by $(\tilde{R}', (\tilde{x}_1, \ldots, \tilde{x}_r))$ the r-boundaried subgraph obtained from $\tilde{R}$ by the deletion of the vertices that are outside of W. By Lemma 4, $R_2 = (R_1', (\tilde{x}_1, \ldots, \tilde{x}_r)) \oplus_{\rm b} (\tilde{R}', (\tilde{x}_1, \ldots, \tilde{x}_r))$ is $\mathcal{F}$-free. Observe also that $\omega'(V(R_2)) \geq \min\{\omega'(V(R_1)), w\}$. It implies that $(G', I', O', \hat{B}', \omega', t')$ has a nonempty solution R_2 with $\omega'(V(R_2)) \geq \min\{\omega'(V(R_1)), w\}$. □

Since the instances $(G, I, O, B, \omega, t, T)$ and $(G, I, O, \hat{B}, \omega, t, T)$ of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph are equivalent, we can consider $(G, I, O, \hat{B}, \omega, t, T)$. Now we apply some reduction rules with the ultimate aim to reduce the size of G. We say that a reduction rule is *safe* if it either produces an equivalent instance of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph or correctly reports that we have no solution. To simplify notation, we keep using $(G, I, O, \hat{B}, \omega, t, T)$ for the instances obtained by the rules.

The rules exploit the following observation.

Claim 4.4. *Let Q be a component of $G[W] - S$. For every nonempty graph R in a solution of $(G, I, O, \hat{B}, w, t, T)$, either $V(Q) \subseteq V(R)$ or $V(Q) \cap V(R) = \emptyset$. Moreover, if $V(Q) \cap V(R) = \emptyset$, then $N_{G[W]}[V(Q)] \cap V(R) = \emptyset$, and if $v \in N_{G[W]}(V(Q))$ is a vertex of R, then $V(Q) \subseteq V(R)$.*

Proof of Claim 4.4. Recall that for every nonempty R in a solution of $(G, I, O, \hat{B}, w, t, T)$, $N_G[V(R)] \subseteq \hat{B}$. Hence, if Q is a component of $G[W] - S$, Q cannot contain two adjacent vertices u and v such that $u \in V(R)$ and $v \notin V(R)$. Therefore, either $V(Q) \subseteq V(R)$ or $V(Q) \cap V(R) = \emptyset$. Similarly, if $V(Q) \cap V(R) = \emptyset$, then $N_{G[W]}[V(Q)] \cap V(R) = \emptyset$, and if $v \in N_{G[W]}(V(Q))$ is a vertex of R, then $V(Q) \subseteq V(R)$. ☐

The following rule is applied for each component Q of $G[W] - S$ exactly once. Claim 4.4 immediately implies that the rule is safe.

Reduction Rule 4.4. For a component Q of $G[W] - S$ do the following in the given order:

- if $N_{G[W]}[V(Q)] \cap I \neq \emptyset$ and $V(Q) \cap O \neq \emptyset$, then return $\emptyset$ and stop,
- if $N_{G[W]}[V(Q)] \cap I \neq \emptyset$, then set $I := I \cup V(Q)$,
- if $V(Q) \cap O \neq \emptyset$, then set $O := O \cup N_{G[W]}[V(Q)]$.

Notice that after application of Reduction Rule 4.4 we have that either $V(Q) \subseteq I$ or $V(Q) \subseteq O$ or $V(Q) \cap (I \cup O \cup \hat{B}) = \emptyset$ for every component Q of $G[W] - S$; the latter property holds since $V(Q) \cap \hat{B} = \emptyset$.

The next rule is applied for all pairs of components Q_1 and Q_2 with $N_{G[W]}(V(Q_1)) = N_{G[W]}(V(Q_2))$ and $|N_{G[W]}(V(Q_1))| = |N_{G[W]}(V(Q_2))| > t$, and for each pair the rule is applied once.

Reduction Rule 4.5. For components Q_1 and Q_2 of $G[W] - S$ such that $N_{G[W]}(V(Q_1)) = N_{G[W]}(V(Q_2))$ and $|N_{G[W]}(V(Q_1))| = |N_{G[W]}(V(Q_2))| > t$ do the following in the given order:

- if $(V(Q_1) \cup V(Q_2)) \cap I \neq \emptyset$ and $(V(Q_1) \cup V(Q_2)) \cap O \neq \emptyset$, then return $\emptyset$ and stop,
- if $(V(Q_1) \cup V(Q_2)) \cap I \neq \emptyset$, then set $I := I \cup (V(Q_1) \cup V(Q_2))$,
- if $(V(Q_1) \cup V(Q_2)) \cap O \neq \emptyset$, then set $O := O \cup N_{G[W]}[V(Q_1) \cup V(Q_2)]$.

Claim 4.5. *Reduction Rule 4.5 is safe.*

Proof of Claim 4.5. Suppose that Q_1 and Q_2 are components of $G[W]-S$ such that $N_{G[W]}(V(Q_1)) = N_{G[W]}(V(Q_2))$ and $|N_{G[W]}(V(Q_1))| = |N_{G[W]}(V(Q_2))| > t$. Then if $V(Q_1) \subseteq V(R)$ for a nonempty graph R in a solution of $(G, I, O, \hat{B}, \omega, t, T)$, then at least one vertex of $N_{G[W]}(V(Q_1))$ is in R, as R has at most t neighbors outside R. Therefore, $V(Q_2) \subseteq V(R)$. Hence, $V(Q_1)$ and $V(Q_2)$ are either both inside R or both outside R. This immediately implies safeness. □

If $V(Q) \subseteq O$ for a component Q of $G[W] - S$, then $N_{G[W]}(V(Q)) \subseteq O$. It immediately implies that the vertices of Q are irrelevant and can be removed. Hence, the following rule is safe.

Reduction Rule 4.6. If there is a component Q of $G[W] - S$ such that $N_{G[W]}(V(Q)) \subseteq O$, then set $G := G - V(Q)$, $W := W \setminus V(Q)$ and $O := O \setminus V(Q)$.

Notice that for each component Q, we have now that either $V(Q) \subseteq I$ or $V(Q) \subseteq W \setminus (I \cup O \cup \hat{B})$.

To define the remaining rules, we construct the sets $\mathcal{G}'_p$ for all $p \in \{0, \ldots, |S|\}$ (see Definition 10) in time $2^{2^{\mathcal{O}(t \log t)}}$ using Lemma 9 and the inequality (2).

We consider inclusion maximal families of components of $G[W]-S$ that have the same neighborhood and replace them by gadgets of bounded size. First, we deal with families whose neighborhoods have size at least $t+1$. We exploit the property that every graph in the solution of $(G, I, O, \hat{B}, \omega, t, T)$ either contains all of the members of each family or none.

Reduction Rule 4.7. Let $L = \{x_1, \ldots, x_p\} \subseteq S$, $p > t$, and let $x = (x_1, \ldots, x_p)$. Let also $Q_1, \ldots, Q_r$ be the components of $G[W] - S$ with $r \geq 1$, and $N_{G[W]}(V(Q_i)) = L$ for all $i \in \{1, \ldots, r\}$. Let $Q = G[\bigcup_{i=1}^{r} N_{G[W]}[V(Q_i)]]$ and $w' = \sum_{i=1}^{r} \omega(V(Q_i))$. Find a p-boundaried graph $(H, y) \in \mathcal{G}'_p$ such that $(H, y) \equiv_{\mathcal{F}_{\mathrm{b}}} (Q, x)$ and denote by A the set of nonboundary vertices of H. Then do the following.

- Delete the vertices of $V(Q_1), \ldots, V(Q_r)$ from G and denote the obtained graph G'.
- Set $G := (G', x) \oplus_{\mathrm{b}} (H, y)$ and $W := (W \setminus \bigcup_{i=1}^{r} V(Q_i)) \cup A$.
- Select arbitrarily $u \in A$ and modify ω as follows:
  - keep the same weight for every $v \in V(G')$ including the boundary vertices $x_1, \ldots, x_p$,
  - set $\omega(v) = 0$ for each $v \in A \setminus \{u\}$,
  - set $\omega(u) = w'$.
- If $V(Q_1) \subseteq I$, then set $I := I \setminus (\bigcup_{i=1}^{r} V(Q_i)) \cup A$.

Claim 4.6. *Reduction Rule 4.7 is safe.*

Proof of Claim 4.6. Denote by $(G', I', O', \hat{B}, \omega', t, T)$ the instance of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained by the application of the rule. Clearly, condition (i) of Definition 12 is fulfilled and we have to verify (ii). For the forward direction, let R be a nonempty graph in the solution of $(G, I, O, \hat{B}, \omega, t, T)$ obtained with respect to some boundary complementation.

If $V(R) \cap V(Q) = \emptyset$, then it is straightforward to see that R is a t-secluded $\mathcal{F}$-free graph for the instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in $(G', I', O', \hat{B}, \omega', t, T)$ for the same boundary complementation. Then condition (a) of Definition 12 (ii) holds.

Suppose that $V(R) \cap V(Q) \neq \emptyset$. By Claim 4.4, there is $i \in \{1, \ldots, r\}$ such that $V(Q_i) \subseteq V(R)$. Since $p > t$, there is $z \in N_{G[W]}[V(Q_i)]]$ such that $z \in V(R)$. Then by Claim 4.4, we conclude that $V(Q_i) \subseteq V(R)$ for all $i \in \{1, \ldots, r\}$. Then $R = (R', x) \oplus_{\mathrm{b}} (Q, x)$, where $R' = R - \bigcup_{i=1}^{r} V(Q_i)$. Let $R'' = (R', x) \oplus_{\mathrm{b}} (H, y)$. Since at least one vertex of x is in R and $H \in \mathcal{G}'_p$, R'' is connected. By Lemma 4, R'' is a t-secluded $\mathcal{F}$-free graph for the instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in $(G', I', O', \hat{B}, \omega', t, T)$ for the same boundary complementation as for R. Because $\omega'(V(R'')) = \omega(R)$, we obtain that condition (a) of Definition 12 (ii) holds.

For the opposite direction, that is, for the proof that condition (b) of Definition 12 (ii) is fulfilled, we use the same arguments using symmetry. □

The rule is applied exactly once for each inclusion maximal set of components $\{Q_1, \ldots, Q_r\}$ having the same neighborhood of size at least $t + 1$.

Our next aim is to analyze components Q of $G[W] - S$ with $|N_{G[W]}(V(Q))| \leq t$. Notice that if we have several components $Q_1, \ldots, Q_r$ of $G[W] - S$ with the same neighborhood $N_{G[W]}(V(Q_i))$ and $|N_{G[W]}(V(Q_i))| \leq t$, then it can happen that there are $i, j \in \{1, \ldots, r\}$ such that $V(Q_i) \subseteq V(R)$ and $N_{G[W]}[V(Q_j)] \cap V(R) = \emptyset$ for R in a solution of $(G, I, O, \hat{B}, \omega, t, T)$. But if $N_{G[W]}[V(Q_j)] \cap V(R) = \emptyset$, then by the connectivity of R and the fact that $G[W] - S$ does not contain boundary terminals, we have that $R = Q_i$. Notice that, in particular, this means that R is a solution for an instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained by the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$. Recall that we output R in this case only if its weight is at least w.

In our next rule, we consider the case when there is Q with $V(Q) \subseteq I$.

Reduction Rule 4.8. Let $L = \{x_1, \ldots, x_p\} \subseteq S$, $p \leq t$, and let $x = (x_1, \ldots, x_p)$. Let also $Q_0, \ldots, Q_r$ be the components of $G[W] - S$ with $r \geq 0$ and $N_{G[W]}(V(Q_i)) = L$ for all $i \in \{0, \ldots, r\}$ and it holds that $V(Q_0) \subseteq I$. Let $Q'_0 = G[N_{G[W]}[V(Q_0)]]$, $Q = G[\bigcup_{i=1}^{r} N_{G[W]}[V(Q_i)]]$ and $w' = \sum_{i=1}^{r} \omega(V(Q_i))$. Find a p-boundaried graph $(H_0, y) \in \mathcal{G}'_p$ such that $(H_0, y) \equiv_{\mathcal{F}_{\mathrm{b}}} (Q_0, x)$ and denote by A_0 the set of nonboundary vertices of H_0, and find a p-boundaried graph $(H, y) \in \mathcal{G}'_p$ such that $(H, y) \equiv_{\mathcal{F}_{\mathrm{b}}} (Q, x)$ and denote by A the set of nonboundary vertices of H. Then do the following.

- Delete the vertices of $V(Q_0), \ldots, V(Q_r)$ from G and denote the obtained graph G'.
- Set $G := (((G', x) \oplus_{\mathrm{b}} (H_0, y)), y) \oplus_{\mathrm{b}} (H, y)$ and $W := (W \setminus \bigcup_{i=0}^{r} V(Q_i)) \cup A_0 \cup A$.
- Select arbitrarily $u \in A_0$ and $v \in A$ and modify ω as follows:
  - keep the same weight for every $z \in V(G')$ including the boundary vertices $x_1, \ldots, x_p$,
  - set $\omega(z) = 0$ for each $z \in (A_0 \setminus \{u\}) \cup (A \setminus \{v\})$,
  - set $\omega(u) = \omega(V(Q_0))$ and $\omega(v) = w'$.
- If $V(Q_i) \subseteq I$ for some $i \in \{1, \ldots, r\}$, then set $I := I \setminus (\bigcup_{i=1}^{r} V(Q_i)) \cup A$.

Claim 4.7. *Reduction Rule 4.8 is safe.*

Proof of Claim 4.7. The claim is proved similarly to Claim 4.6. Let $(G', I', O', \hat{B}, \omega', t, T)$ be the instance of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained by the application of the rule. As condition (i) of Definition 12 is trivial, we have to verify (ii).

For the forward direction, let R be a nonempty graph in the solution of $(G, I, O, \hat{B}, \omega, t, T)$ obtained with respect to some boundary complementation.

Suppose that $L \cap V(R) = \emptyset$. Then $R = Q_0$ and R is a solution for an instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained by the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$. Let $R' = H_0$. By Lemma 4, we have that R' is $\mathcal{F}$-free, that is, R' is a t-secluded $\mathcal{F}$-free graph for the instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in $(G', I', O', \hat{B}, \omega', t, T)$ for the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$. Because $\omega'(V(R')) = \omega(R)$, condition (a) of Definition 12 (ii) holds.

Assume now that $L \cap V(R) \neq \emptyset$. By Claim 4.4, $V(Q_i) \subseteq V(R)$ for all $i \in \{0, \ldots, r\}$. Then we can write that $R = ((R', x) \oplus_{\mathrm{b}} (Q_0', x), x) \oplus_{\mathrm{b}} (Q, x)$, where $R' = R - \bigcup_{i=0}^{r} V(Q_i)$. Let $R'' = ((R', x) \oplus_{\mathrm{b}} (H_0, y), y) \oplus_{\mathrm{b}} (H, y)$. Then by the same arguments as in the proof of Claim 4.6, we conclude that R'' is a t-secluded $\mathcal{F}$-free graph for the instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in $(G', I', O', \hat{B}, \omega', t, T)$ for the same boundary complementation as for R. Because $\omega'(V(R'')) = \omega(R)$, we obtain that condition (a) of Definition 12 (ii) holds.

For the opposite direction, that is, for the proof that condition (b) of Definition 12 (ii) is fulfilled, we use the same arguments performing the opposite replacements in a graph from the solution of $(G', I', O', \hat{B}, \omega', t, T)$. $\square$

Reduction Rule 4.8 is applied exactly once for each inclusion maximal set of components $\{Q_1, \ldots, Q_r\}$ having the same neighborhood of size at most t such that at least one of the components contains a vertex of I.

Finally, we consider inclusion maximal families of components of $G[W] - S$ that have the same neighborhood of size at most t such that there is no component in the family whose vertices are in I.

Reduction Rule 4.9. Let $L = \{x_1, \ldots, x_p\} \subseteq S$, $p \leq t$, and let $x = (x_1, \ldots, x_p)$. Let also $Q_0, \ldots, Q_r, r \geq 0$, be the components of $G[W] - S$ with $N_{G[W]}(V(Q_i)) = L$ for all $i \in \{0, \ldots, r\}$ such that $\omega(V(Q_0)) \geq \omega(V(Q_i))$ for every $i \in \{1, \ldots, r\}$ and the p-boundaried graphs $(G[N_{G[W]}[V(Q_i)]], (x_1, \ldots, x_p))$ are pairwise equivalent with respect to $\mathcal{F}_{\mathrm{b}}$ for all $i \in \{0, \ldots, r\}$. Let $Q'_0 = G[N_{G[W]}[V(Q_0)]]$, $Q = G[\bigcup_{i=1}^{r} N_{G[W]}[V(Q_i)]]$, and $w' = \min\{w - 1, \sum_{i=1}^{r} \omega(V(Q_i))\}$. Find a p-boundaried graph $(H_0, y) \in \mathcal{G}'_p$ such that $(H_0, y) \equiv_{\mathcal{F}_{\mathrm{b}}} (Q_0, x)$ and denote by A_0 the set of nonboundary vertices of H_0, and find a p-boundaried graph $(H, y) \in \mathcal{G}'_p$ such that $(H, y) \equiv_{\mathcal{F}_{\mathrm{b}}} (Q, x)$ and denote by A the set of nonboundary vertices of H. Then do the following.

- Delete the vertices of $V(Q_0), \ldots, V(Q_r)$ from G and denote the obtained graph G'.
- Set $G := (((G', x) \oplus_{\mathrm{b}} (H_0, y)), y) \oplus_{\mathrm{b}} (H, y)$ and $W := (W \setminus \bigcup_{i=0}^{r} V(Q_i)) \cup A_0 \cup A$.
- Select arbitrarily $u \in A_0$ and $v \in A$ and modify ω as follows:
  - keep the same weight for every $z \in V(G')$ including the boundary vertices $x_1, \ldots, x_p$,
  - set $\omega(z) = 0$ for each $z \in (A_0 \setminus \{u\}) \cup (A \setminus \{v\})$,
  - set $\omega(u) = \omega(V(Q_0))$ and $\omega(v) = w'$.

Claim 4.8. *Reduction Rule 4.9 is safe.*

Proof of Claim 4.8. We use the same approach as in the proofs of Claims 4.6 and 4.7. Let $(G', I', O', \hat{B}, \omega', t, T)$ be the instance of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained by the application of the rule. As in the aforementioned claims, it is sufficient to verify condition (ii) of Definition 12.

For the forward direction, let R be a nonempty graph in the solution of $(G, I, O, \hat{B}, \omega, t, T)$ obtained with respect to some boundary complementation.

If $V(R) \cap (\bigcup_{i=0}^{r} V(Q_i)) = \emptyset$, then R is a t-secluded $\mathcal{F}$-free graph for the instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in $(G', I', O', \hat{B}, \omega', t, T)$ for the same boundary complementation. Then condition (a) of Definition 12 (ii) holds.

Assume that there are $i, j \in \{0, \ldots, r\}$ such that $V(R) \cap V(Q_i) \neq \emptyset$ and $V(R) \cap V(Q_j) = \emptyset$. Then $R = Q_i$ and R is a solution for an instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained by the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$. Let $R' = H_0$. Since $(Q'_0, x) \equiv_{\mathcal{F}_{\mathrm{b}}} (H_0, y)$, R' is $\mathcal{F}$-free by Lemma 4, that is, R' is a t-secluded $\mathcal{F}$-free graph for the instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in $(G', I', O', \hat{B}, \omega', t, T)$ for the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$. Because $\omega'(V(R')) = \omega(V(Q_i)) \leq \omega(V(Q_0)) = \omega(V(R))$, we obtain that condition (a) of Definition 12 (ii) holds.

Suppose now that $V(Q_i) \subseteq V(R)$ for every $i \in \{0, \ldots, r\}$. Note that since R is connected, $L \cap V(R) \neq \emptyset$. Then $R = ((R', x) \oplus_{\mathrm{b}} (Q_0, x), x) \oplus_{\mathrm{b}} (Q, x)$, where $R' = R - \bigcup_{i=0}^{r} V(Q_i)$. Let $R'' = ((R', x) \oplus_{\mathrm{b}} (H_0, y), y) \oplus_{\mathrm{b}} (H, y)$. Then we conclude that R'' is a connected t-secluded $\mathcal{F}$-free graph for the instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in $(G', I', O', \hat{B}, \omega', t, T)$ for the same boundary complementation as for R. Because $\omega'(V(R'')) \geq \min\{w, \omega(R)\}$, we obtain that condition (a) of Definition 12 (ii) holds.

For the opposite direction, that is, for the proof that condition (b) of Definition 12 (ii) is fulfilled, we use similar arguments performing the opposite replacements in a graph from the solution of $(G', I', O', \hat{B}, \omega', t, T)$. The difference from the proofs of Claims 4.6 and 4.7 is that now we have no complete symmetry. Let R' be a nonempty graph in the solution of $(G', I', O', \hat{B}, \omega', t, T)$ obtained with respect to some boundary complementation. For the cases when $(A_0 \cup A) \cap V(R') = \emptyset$ or $A_0 \cup A \subseteq V(R')$, the arguments are the same. Suppose that $A_0 \cup A$ contain a vertex of R' and a vertex that is not in R'. Because $\omega'(A) \leq w - 1$, we obtain that $R' = G'[A_0]$ and R' is a solution for an instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained by the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$. Then we conclude that $R = Q_0$ is a t-secluded $\mathcal{F}$-free graph for the instance of w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph arising in $(G, I, O, \hat{B}, \omega, t, T)$ for the boundary complementation with respect to $(\emptyset, \emptyset, T, \emptyset)$. $\square$

The Reduction Rule 4.9 is applied for each inclusion maximal set of components $\{Q_1, \ldots, Q_r\}$ satisfying the conditions of the rule.

Our next aim is to upper bound the size of the graph obtained by the reductions. Denote by $(G^*, I^*, O^*, B^*, \omega^*, t, T)$ the instance of Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph obtained from $(G, I, O, \hat{B}, \omega, t, T)$ by Reduction Rules 4.4-4.9. Notice that all modifications were made for $G[W]$. Denote by W^* the set of vertices of the graph obtained from the initial $G[W]$ by the rules.

Claim 4.9.

$$|W^*| \leq 2^{|S|} 2|S|^c t^c + |S|. \tag{3}$$

Proof of Claim 4.9. Observe that there are at most $2^{|S|}$ subsets L of S such that there is a component Q of $G[W] - S$ with $N_{G[W]}(V(Q)) = L$. If $|L| > t$, then all Q with $N_{G[W]}(V(Q)) = L$ are replaced by one graph by Reduction Rule 4.7 and the number of vertices of this graph is at most $|L|^c$ by the choice of the constant c. If $|L| \leq t$, then we either apply Reduction Rule 4.8 for all Q with $N_{G[W]}(V(Q)) = L$ and replace these components by two graphs with at most $|L|^c$ vertices or we apply Reduction Rule 4.9. For the latter case, observe that there are at most t^c partitions of the components Q with $N_{G[W]}(V(Q)) = L$ into equivalence classes with respect to $\mathcal{F}_{\mathrm{b}}$ by Lemma 7. Then we replace each class by two graphs with at most $|L|^c$ vertices. Taking into account the vertices of S, we obtain the bound $|W^*| \leq 2^{|S|}2|S|^c t^c + |S|$. $\square$

By (1) and (2), $|W^*| \leq q$. Recall that $|W \setminus U| > q$. Therefore, $|V(G^*)| < |V(G)|$. We use it and solve Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for $(G^*, I^*, O^*, B^*, \omega^*, t, T)$ recursively.

Our final task is to evaluate the running time. Denote by $\tau(G, I, O, B, \omega, t, T)$ the time needed to solve Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for $(G, I, O, B, \omega, t, T)$. Lemmas 5 and 7 imply that the reduction rules are polynomial. The algorithm from Lemma 3 runs in time $2^{2^{\mathcal{O}(t \log t)}} \cdot n^{\mathcal{O}(1)}$. Notice that the sets $\mathcal{G}_p$ and $\mathcal{G}'_p$ can be constructed separately from running the algorithm for Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph. Then we obtain the following recurrence for the running time:

$$\begin{aligned}\tau(G, I, O, B, \omega, t, T) \leq & \tau(G^*, I^*, O^*, B^*, \omega^*, t, T) \\ & + \tau(\tilde{G}, \tilde{I}, \tilde{O}, \tilde{B}, \tilde{\omega}, t, \tilde{T}) + 2^{2^{\mathcal{O}(t \log t)}} \cdot n^{\mathcal{O}(1)}.\end{aligned} \tag{4}$$

Note that $|V(\tilde{G})| > q$, because $V(\tilde{G}) = W$ and $|W \setminus U| > q$. Because $|W^*| \leq q$,

$$|V(G^*)| \leq |V(G)| - |V(\tilde{G})| + q. \tag{5}$$

Recall that if the algorithm of Lemma 3 reports that G is $((2q+1)q \cdot 2^t, t)$-unbreakable or we have that $|V(G)| \leq (2q+1)q \cdot 2^t$, we do not recurse but solve the problem directly in time $2^{2^{2^{\mathcal{O}(t \log t)}}} \cdot n^{\mathcal{O}(1)}$. Following the general scheme from [5], we obtain that these conditions together with (4) and (5) imply that the total running time is $2^{2^{2^{\mathcal{O}(t \log t)}}} \cdot n^{\mathcal{O}(1)}$. $\square$

We have now all the details in place to be able to prove Theorem 1 that we restate.

Theorem 1. Connected Secluded $\mathcal{F}$-Free Subgraph *can be solved in time* $2^{2^{2^{\mathcal{O}(t \log t)}}} \cdot n^{\mathcal{O}(1)}$.

Proof. Let (G, ω, t, w) be an instance of Connected Secluded $\mathcal{F}$-Free Subgraph. We set $I = \emptyset$, $O = \emptyset$, $B = V(G)$ and $T = \emptyset$. Then we solve Boundaried w-Maximum Connected Secluded $\mathcal{F}$-Free Subgraph for (G, I, O, B, w, t, T) using Lemma 14 in time $2^{2^{2^{\mathcal{O}(t \log t)}}} \cdot n^{\mathcal{O}(1)}$. It remains to notice that (G, ω, t, w) is a yes-instance of Connected Secluded $\mathcal{F}$-Free Subgraph if and only if $(G, I, O, B, \omega, t, T)$ has a nonempty graph in a solution. $\square$

5. Algorithms for special cases of Connected Secluded Π-Subgraph

We applied the recursive understanding technique introduced by Chitnis et al. [5] for Connected Secluded Π-Subgraph when Π is defined by a finite set of forbidden subgraphs. The drawback of applying the recursive understanding technique for this problem is that we got triple-exponential dependence on the parameter in our FPT algorithms. It is natural to ask whether we can do better for some properties Π. In this section we show that it can be done if Π is the property to be a complete graph, a star, to be d-regular or to be a path.

5.1. Secluded Clique

Recall that a *clique* is a set of pairwise adjacent vertices. We begin with the Secluded Clique problem, defined as follows.

Secluded Clique	
Input:	A graph G, a weight function $\omega\colon V(G) \to \mathbb{Z}_{>0}$, a nonnegative integer t and a positive integer w.
Task:	Decide whether G contains a t-secluded clique U with $\omega(U) \geq w$.

We prove that this problem can be solved in time $2^{\mathcal{O}(t^2)} \cdot n^{\mathcal{O}(1)}$. The result uses the algorithm of Lemma 2 and the following simple observations.

Lemma 15. *Let (G, ω, t, w) be an input of* Secluded Clique *and let U be an inclusion maximal solution, that is, a t-secluded clique with $\omega(U) \geq w$. Let L be an inclusion maximal set of true twins of G. Then $L \cap U \neq \emptyset$ implies that $L \subseteq U$.*

Proof. Let L be an inclusion maximal set of true twins, and let $u, v \in L$ be such that $u \in U$ and $v \notin U$. Consider $U' = G[U \cup \{v\}]$. Since U is a t-secluded clique, and u, v are true twins, we have that U' is also a t-secluded clique, and $\omega(U') = \omega(U) + \omega(v) \geq w$. Therefore U' is also a solution of Secluded Clique, contradicting the maximality of U. □

Let $\mathcal{L}$ be the family of all maximal sets of true twins in a graph G. Note that a vertex can not belong to two different maximal sets of true twins, so $\mathcal{L}$ induces a partition of $V(G)$. Consider the graph $\tilde{G}$ obtained from G by contracting each maximal set of true twins L into a single vertex x_L. In other words, $\tilde{G}$ contains one vertex for each element of $\mathcal{L}$. Two vertices x_1 and x_2 in $\tilde{G}$ are adjacent if there is an edge in G with one endpoint in L_1 and the other one in L_2, where L_1 and L_2 are elements of $\mathcal{L}$ corresponding to x_1 and x_2, respectively.

We say that a vertex $x \in \tilde{G}$ is a *contraction* of $L \in \mathcal{L}$ if x is the vertex of $\tilde{G}$ corresponding to L. We say that a set $U \subseteq V(G)$ is the *expansion* of $\tilde{U} \subseteq V(\tilde{G})$ if $U = \bigcup_{x \in \tilde{U}} L_x$, where x is the contraction of $L_x \in \mathcal{L}$. We also say in that case that $\tilde{U}$ is the *contraction* of U.

Lemma 16. *Let G be a graph, t be a positive integer and let U be an inclusion maximal t-secluded clique in G. There exists a set $\tilde{U}$ of vertices of $V(\tilde{G})$ such that:*

1) U is the expansion of $\tilde{U}$, *2) $\tilde{U}$ is a t-secluded clique on $\tilde{G}$, and*

3) $|\tilde{U}| \le 2^t$.

Proof. Let $\tilde{U} \subseteq V(\tilde{G})$ be the set of vertices that are contractions of the maximal sets of true twins in G intersecting U. We claim that $\tilde{U}$ satisfies the desired properties.

1) From Lemma 15, we know that if a maximal set of true twins L intersects U, then $L \subseteq U$. Therefore $\tilde{U}$ is a contraction of U (so U is an expansion of $\tilde{U}$).

2) Let x_1 and x_2 be two vertices in $\tilde{U}$ that are contractions of L_1 and L_2, respectively. Since U is a clique in G, L_1 and L_2 must contain adjacent vertices, so x_1 and x_2 are adjacent in $\tilde{G}[\tilde{U}]$. Moreover, $|N_{\tilde{G}}(\tilde{U})|$ equals the number of maximal sets of true twins intersecting $N_G(U)$, so $|N_{\tilde{G}}(\tilde{U})| \le |N_G(U)| \le t$. We conclude that $\tilde{U}$ induces a t-secluded clique in $\tilde{G}$.

3) Let x_1 and x_2 be two different vertices in $\tilde{U}$ that are contractions of L_1 and L_2, respectively. From the definition of maximal sets of true twins, $N_G(L_1) \neq N_G(L_2)$, so $N_{\tilde{G}}(x_1) \neq N_{\tilde{G}}(x_2)$. Since $\tilde{U}$ is a clique, necessarily $N_{\tilde{G}}(x_1) \cap \tilde{U} = N_{\tilde{G}}(x_2) \cap \tilde{U}$. Therefore, every vertex in $\tilde{U}$ has a different neighborhood outside $\tilde{U}$. Since $|N_{\tilde{G}}(\tilde{U})| \le t$, we obtain that $|\tilde{U}| \le 2^{|N_{\tilde{G}}(\tilde{U})|} \le 2^t$.

□

Theorem 3. Secluded Clique *can be solved in time $2^{\mathcal{O}(t^2)} \cdot n^{\mathcal{O}(1)}$.*

Proof. The algorithm for Secluded Clique on input (G, ω, t, w) first computes the family $\mathcal{L}$ of all inclusion maximal set of true twins of G, and then computes $\tilde{G}$. Note that this can be done in linear time (see, e.g., [19]). Then, the algorithm uses Lemma 2 to compute in time $2^{\mathcal{O}(t^2)} n \log n$ a family $\mathcal{S}$ of at most $2^{\mathcal{O}(t^2)} \log n$ subsets of $V(\tilde{G})$ such that: for any sets $A, B \subseteq V(\tilde{G})$, $A \cap B = \emptyset$, $|A| \le 2^t$, $|B| \le t$, there exists a set $S \in \mathcal{S}$ with $A \subseteq S$ and $B \cap S = \emptyset$.

Let U be a set of vertices of G inducing an inclusion maximal solution of Secluded Clique on instance (G, ω, t, w), and let $\tilde{U}$ be the contraction of U. From Lemma 16, we know that $|N_{\tilde{G}}(\tilde{U})| \le t$ and $|\tilde{U}| \le 2^t$. Then, there exists $S \in \mathcal{S}$ such that $\tilde{U} \subseteq S$ and $N_{\tilde{G}}(\tilde{U}) \cap S = \emptyset$. In other words $\tilde{U}$ is a component of $\tilde{G}[S]$. Therefore, by checking every $S \in \mathcal{S}$ and every component C of $\tilde{G}[S]$ we find a secluded clique if it exists. □

5.2. Secluded Star

Another example of a particular problem where we have a better running time is Secluded Star. For a positive integer r, call $K_{1,r}$ the complete bipartite graph with one vertex in one part and r vertices in the other part. A tree isomorphic to $K_{1,r}$ is called a *star*.

Secluded Star

Input:	A graph G, a weight function $\omega\colon V(G) \to \mathbb{Z}_{>0}$, a nonnegative integer t, and a positive integer w.
Task:	Decide whether G contains a t-secluded induced star S with $\omega(V(S)) \geq w$.

In this case, a faster FPT algorithm can be deduced via a reduction to the problem Vertex Cover parameterized by the size of the solution.

For a graph G and x in $V(G)$, we denote by $N_G^2(x)$ the set of vertices at distance 2 from x, i.e., $N_G^2(x)$ is the set of vertices $u \in V(G)$ such that $u \notin N_G[x]$ and there exists $v \in N_G(x)$ such that $u \in N_G(v)$. We also call $N_G^2[x]$ the set $N_G^2(x) \cup N_G[x]$. Let now $F_x = (N_G^2(x) \cup N_G(x), E')$ be the subgraph of $G[N_G^2(x) \cup N_G(x)]$ such that $E' = E(G[N_G^2(x) \cup N_G(x)]) \setminus E(G[N_G^2(x)])$, i.e., F_x is the graph induced by the vertices in $N_G^2(x) \cup N_G(x)$ after the deletion of all edges between vertices in $N_G^2(x)$. Note that x is not a vertex of F_x.

A vertex x is the center of a star S if x is the vertex of maximum degree in S. The following lemma relates the center x of a t-secluded star S of a graph G with a vertex cover of size at most t of F_x.

Lemma 17. *Let S be a t-secluded star on a graph G with a center x. Then $N_G(V(S))$ is a vertex cover of F_x. Moreover, if S is an inclusion maximal t-secluded star with a center x, then $N_G(V(S))$ is an inclusion minimal vertex cover of F_x.*

Proof. Let u, v be two adjacent vertices of F_x. Since $F_x[N^2(x)]$ is edgeless, we assume w.l.o.g. that u belongs to $N_G(x)$. If u is contained in $N_G(x) \setminus V(S)$, then u is in $N(V(S))$ (because x belongs to S, so $N_G(x) \setminus V(S) \subseteq N_G(V(S))$). If u is in S, then either v is in $N_G^2(x)$ or v is in $N_G(x) \setminus V(S)$ (because $F_x[V(S)]$ is edgeless). In both cases v is in $N_G(V(S))$. We conclude that either u or v is contained in $N_G(V(S))$.

Assume that S is an inclusion maximal t-secluded star with the center x. Suppose that, contrary to the second claim, $N_G(V(S))$ is not an inclusion minimal vertex cover of F_x, that is, there is $u \in N_G(V(S))$ such that $X = N_G(V(S)) \setminus \{u\}$ is a vertex cover of F_x. Because X is a vertex cover of F_x and $X \subseteq N_G(V(S))$, we have that $u \in N_G(x)$ and $N_G(u) \setminus \{x\} \subseteq X$. It implies that $S' = G[V(S) \cup \{u\}]$ is a t-secluded star contradicting the maximality of S. $\square$

A basic result on parameterized complexity states that one can decide whether a graph contains a vertex cover of size at most t by branching algorithms. These algorithms can be adapted to output the list of all inclusion minimal vertex covers of size at most t within the same running time, and we immediately obtain the following claim.

Proposition 1 ([6, 8])**.** *There is an algorithm computing the list of all the inclusion minimal vertex covers of size at most t of a graph G in time $\mathcal{O}(2^t(n+m))$.*

Theorem 4. Secluded Star *can be solved in time* $2^t \cdot n^{\mathcal{O}(1)}$.

Proof. Let (G, ω, t, w) be an input of Secluded Star. The algorithm starts computing, for every $x \in V(G)$, the list of all inclusion minimal vertex covers of size at most t of F_x using Proposition 1. Then, for every vertex cover U of size at most t of F_x, the algorithm checks if $N_G[x] \setminus U$ induces in G a solution of the problem. We know from Lemma 17, that if S is a solution of Secluded Star on input (G, ω, t, w), then $N_G(V(S))$ is an inclusion minimal vertex cover of size at most t of F_x. Note also that S is an induced star with center x in a graph G, then $V(S) = N_G[x] \setminus N_G(V(S))$. □

5.3. Secluded Regular Subgraph

Our next example is Connected Secluded Regular Subgraph. For a positive integer d, a graph G is *d-regular* if $d_G(v) = d$ for every $v \in V(G)$.

Connected Secluded Regular Subgraph

Input:	A graph G, a weight function $\omega\colon V(G) \to \mathbb{Z}_{>0}$, a nonnegative integer t, and positive integers w and d.
Task:	Decide whether G contains a connected t-secluded d-regular induced subgraph H with $\omega(V(H)) \geq w$.

Let (G, ω, t, w, d) be an input of Connected Secluded Regular Subgraph and let U be a set of vertices of G such that $G[U]$ is a solution of the problem. Note first that any vertex of degree greater than $t+d$ cannot be contained in U, otherwise $G[U]$ is not t-secluded. Let $W = \{x \in V(G) \mid d_G(x) \geq t+d+1\}$. If $N(U) \subseteq W$, then U is a component of $G - W$. Therefore, our algorithm will first compute the components of $G-W$ and check if some of them is a solution. In the following we assume that this is not the case, that is, $N(U) \setminus W \neq \emptyset$.

Let $L = N(U) \setminus W$, $U_1 = N(L) \cap U$, $U_2 = N(U_1) \cap U$ and $\tilde{U} = U_1 \cup U_2$ as it is shown in Fig. 3. Note that U_1 separates U and L in $G - W$. Observe that the sets have bounded size as we have that $|L| \leq t$, $|U_1| \leq t \cdot (t+d)$ and $|U_2| \leq dt \cdot (t+d)$, because $U \cap W = \emptyset$. Therefore, $|\tilde{U}| \leq t(d+1)(t+d)$. The main idea of our algorithm is to identify $\tilde{U}$ and L in the input graph.

We need the following definition. A set of vertices C is called *good* for U if

- $C \subseteq U$, and
- for all $u \in C \cap U_1$, $N(u) \cap U \subseteq C$.

Note that every vertex u in a good set C satisfies $|N(u) \cap C| \leq d$. Moreover, if $u \in U_1 \cap C$ then $|N(u) \cap C| = d$. Note also that if C_1 and C_2 are good for U then $C_1 \cup C_2$ is good for U.

Lemma 18. *Let S be a set of vertices of $\tilde{G} = G - W$ satisfying $\tilde{U} \subseteq S$ and $S \cap L = \emptyset$. Let now C be a component of $G[S]$ such that $C \cap U \neq \emptyset$, then:*

1) C is good for U, and

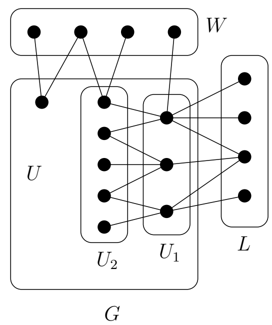

Figure 3: Structure of the sets U, W, L, U_1 and U_2.

2) if $u \in C$ is such that $|N_{\tilde{G}}(u) \cap C| < d$, then $S' = S \cup N_{\tilde{G}}(u)$ satisfies $\tilde{U} \subseteq S'$ and $S' \cap L = \emptyset$.

Proof.

1) Let $u \in C \cap U$ and $v \in N_{\tilde{G}}(u) \cap (C \setminus U)$. Then v is contained in L, which contradicts the fact that $S \cap L = \emptyset$. Therefore $C \subseteq U$. Moreover, if $u \in U_1 \cap C$ then $N_{\tilde{G}}(u) \cap U \subseteq C$, because U_2 is contained in S and C is a connected component of $G[S]$. We conclude that C is good for U.

2) Let $u \in C$ be such that $|N_{\tilde{G}}(u) \cap C| < d$. Since C is good for U we know that u is not contained in U_1, so $N_{\tilde{G}}(u) \cap L = \emptyset$. Therefore $\tilde{U} \subseteq S \subseteq S'$ and $S' \cap L = \emptyset$.

□

Theorem 5. Connected Secluded Regular Subgraph *can be solved in time $2^{\mathcal{O}(t \log(td))} \cdot n^{\mathcal{O}(1)}$.*

Proof. The algorithm for Connected Secluded Regular Subgraph on input (G, ω, t, w, d) first computes the set $W = \{x \in V(G) \mid d_G(x) \geq d + t + 1\}$. Then it computes the connected components of $G - W$ and checks if some of them is a solution. If a solution is not found this way, the algorithm computes $\tilde{G} = G - W$. Then, the algorithm uses Lemma 2 to compute in time $2^{\mathcal{O}(t \log(td))} n \log n$ a family $\mathcal{S}$ of at most $2^{\mathcal{O}(t \log(td))} \log n$ subsets of $V(\tilde{G})$ such that: for any sets $A, B \subseteq V(\tilde{G})$, $A \cap B = \emptyset$, $|A| \leq t(d+1)(t+d)$, $|B| \leq t$, there exists a set $S \in \mathcal{S}$ with $A \subseteq S$ and $B \cap S = \emptyset$.

For each set $S \in \mathcal{S}$, the algorithm marks as *candidate* every component C of $\tilde{G}[S]$ that satisfies that for all $u \in C$, $|N_{\tilde{G}}(u) \cap C| \leq d$. For each candidate component C, the algorithm looks for a vertex u in C such that $|N_{\tilde{G}}(u) \cap C| < d$. If such vertex is found, the corresponding component is enlarged adding to C all vertices in $N_{\tilde{G}}(u)$. If $N_{\tilde{G}}(u)$ intersects other components of $G[S]$, we merge them into C. We repeat the process on the enlarged component C until it can not grow any more, or some vertex u in C satisfies $|N_{\tilde{G}}(u) \cap C| > d$. In the first case we check if the obtained component is a solution of the problem, otherwise the component is unmarked (is not longer a candidate), and the algorithm continues with another candidate component of $\tilde{G}[S]$ or other set $S' \in \mathcal{S}$.

Let U be a set of vertices such that $G[U]$ is a solution of Connected Secluded Regular Subgraph on input (G, ω, t, w, d) with $N(U) \setminus W \neq \emptyset$. From the construction of $\mathcal{S}$, we know that there exists some $S \in \mathcal{S}$ such that $\tilde{U} \subseteq S$ and $S \cap L = \emptyset$. From Lemma 18 (1), we know that any component of $\tilde{G}[S]$ intersecting U is good for U, so it will be marked as candidate. Finally, from Lemma 18 (2) we know that when a component that is good for U grows, the obtained component is also good for U. Indeed, if C is a good component of $\tilde{G}[S]$ containing a vertex u such that $|N_{\tilde{G}}(u) \cap C| < d$, then the component of $\tilde{G}[S \cup N_{\tilde{G}}(u)]$ containing C is also good for U. We conclude that we correctly find U testing the enlarging process on each component of $\tilde{G}[S]$, for each $S \in \mathcal{S}$. $\square$

5.4. Secluded Heavy Path

The same approach can be used for some other problems. In particular, we can do it for Secluded Heavy Path, defined as follows.

Secluded Heavy Path	
Input:	A graph G, a weight function $\omega\colon V(G) \to \mathbb{Z}_{>0}$, a nonnegative integer t, and a positive integer w.
Task:	Decide whether G contains a t-secluded induced path P with $\omega(V(P)) \geq w$.

Corollary 2. Secluded Heavy Path *can be solved in time* $2^{\mathcal{O}(t \log t)} \cdot n^{\mathcal{O}(1)}$.

Proof. Observe that a path is an "almost" 2-regular graph as at most two vertices of a path can have degrees one or zero. We can use this and reduce Secluded Heavy Path to Connected Secluded Regular Subgraph for $d = 2$.

Let (G, ω, t, w) be an instance of Secluded Heavy Path.

First, we check whether the instance has a trivial solution, that is, a path with one vertex. In other words, we check whether there is $v \in V(G)$ with $d_G(v) \leq t$ and $\omega(v) \geq w$. Clearly, this can be done in polynomial time. From now we assume that there is no solution of this type.

Let $s = \sum_{v \in V(G)} \omega(v)$ and $w' = w + s$. For every two distinct vertices $x, y \in V(G)$, we consider the graph $G'_{x,y}$ obtained by adding a new vertex u and making it adjacent to x and y. We define

$$\omega'(z) = \begin{cases} \omega(z) & \text{if } z \in V(G), \\ s & \text{if } z = u. \end{cases}$$

We claim that (G, ω, t, w) is a yes-instance of Secluded Heavy Path if and only if there are distinct $x, y \in V(G)$ such that $(G'_{x,y}, \omega', t, w', 2)$ is a yes-instance of Connected Secluded Regular Subgraph.

Assume that (G,ω,t,w) is a yes-instance of Secluded Heavy Path. Let P be a solution, that is, P is a t-secluded path in G with $\omega(V(P)) \geq w$. Recall, that we assumed that (G,ω,t,w) has no trivial solution. Therefore, P is a path with at least two vertices. Let x and y be the end-vertices of P. Consider the cycle C in $G'_{x,y}$ obtained from P by making u adjacent to x and y. Clearly, $N_{G'_{x,y}}(V(C)) = N_G(V(P))$ and $\omega'(V(C)) = \omega(V(P)) + s \geq w'$. This implies that $(G'_{x,y},\omega',t,w',2)$ is a yes-instance of Connected Secluded Regular Subgraph.

Suppose that $(G'_{x,y},\omega',t,w',2)$ is a yes-instance of Connected Secluded Regular Subgraph for some $x,y \in V(G)$. Let C be a solution, that is, a t-secluded cycle (a connected 2-regular subgraph) with $\omega'(V(C)) \geq w'$. By the definition of s and w', we obtain that $u \in V(C)$. Let x and y be the neighbors of u in C and define P to be the (x,y)-path in C avoiding u. We have that $N_{G'_{x,y}}(V(C)) = N_G(V(P))$ and $\omega(V(P)) = \omega'(V(C)) - \omega'(u) \geq w' - s \geq w$. Therefore, P is a solution for (G,ω,t,w) and we have a yes-instance of Secluded Heavy Path.

Clearly, the constructed Turing reduction of Secluded Heavy Path to Connected Secluded Regular Subgraph is polynomial. Then, by Theorem 5, we conclude that Secluded Heavy Path can be solved in time $2^{\mathcal{O}(t \log t)} \cdot n^{\mathcal{O}(1)}$. □

6. Concluding remarks

We proved that when Π is defined by a finite set of forbidden subgraphs, then Connected Secluded Π-Subgraph (i.e., Connected Secluded $\mathcal{F}$-Free Subgraph) is FPT when parameterized by t. It is natural to ask whether it is possible to extend this result for other interesting graph properties.

We observe that the meta-algorithmic results of Lokshtanov et al. [16] allow to do it for some cases. In particular, they showed the existence of a (nonconstructive) FPT algorithm for Π defined as follows for an integer constant c and a CMOS formula φ: a graph G satisfies Π if and only if the *treewidth* of G is at most c and $G \models \varphi$ (we refer to [6] for the treewidth definition). Notice that the treewidth bound imposes a very strong constraint and it makes the problem relatively easy for unbreakable graphs. Is it possible to relax it?

Recall that the meta-theorem of Lokshtanov et al. [16] gives only an existential result. To obtain a constructive algorithm based on the recursive understanding techniques with an explicit running time, one has to go through the recursion step. In particular, we did it for the case when Π is the property to be acyclic or, in other words, for the Secluded Tree problem, where the task is to find a t-secluded induced subtree of weight at least w. Using the scheme similar to the one used in Section 4 for Connected Secluded $\mathcal{F}$-Free Subgraph, we were able to show that Secluded Tree can be solved in time $2^{2^{\mathcal{O}(t \log t)}} \cdot n^{\mathcal{O}(1)}$. Since the general idea of the algorithm is the same as for Connected Secluded $\mathcal{F}$-Free Subgraph, we omit the proof. Notice that the running time is double-exponential and recall that the running time for Connected Secluded $\mathcal{F}$-Free Subgraph is even worse. Is it possible to improve these running times or/and establish algorithmic lower bounds showing that, say, a double-exponential dependence is unavoidable up to some reasonable complexity assumptions?

Our result for Connected Secluded $\mathcal{F}$-Free Subgraph and the remark for Secluded Tree also indicate that it could be an interesting and challenging problem to classify the parameterized complexity for Connected Secluded Π-Subgraph for hereditary properties that cannot be defined by a finite set of forbidden induced subgraphs. For example, what can be said about Connected Secluded Chordal Subgraph or Connected Secluded Interval Subgraph?

Another interesting question concerns the kernelization for Connected Secluded Π-Subgraph. We refer to [6] for a broader introduction to kernelization algorithms. Recall that 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) $|x'| + k'$ 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. A kernel is *polynomial* if f is polynomial.

For Connected Secluded Π-Subgraph, we hardly can hope to obtain polynomial kernels as it could be easily proved by applying the results of Bodlaender et al. [3] that, unless NP $\subseteq$ coNP /poly, Connected Secluded Π-Subgraph has no polynomial kernel when parameterized by t if Connected Secluded Π-Subgraph is NP-complete. Nevertheless, Connected Secluded Π-Subgraph can have a polynomial *Turing kernel*, defined as follows.

Let $f: \mathbb{N} \to \mathbb{N}$. A *Turing kernelization* of size F for a parameterized problem is an algorithm that decides whether a given instance (x, k) of the problem, where x is an input and k is a parameter, is a yes-instance in time polynomial in $|x| + k$, when given the access to an oracle that decides whether an instance (x', k'), where $|x'| + k' \leq f(k)$, is a yes-instance in a single step. A Turing kernel is polynomial if f is a polynomial.

We show that Connected Secluded Π-Subgraph has a polynomial Turing kernel if Π is the property to be a star.

Theorem 6. Secluded Star *problem admits a polynomial Turing kernelizaton.*

Proof. Let S be a solution of Secluded Star on input (G, ω, t, w). Remember that for each $x \in V(G)$, we called F_x the subgraph of $G[N_G(x) \cup N_G^2(x)]$ obtained by the deletion of all edges with both endpoints in $N_G^2(x)$. From Lemma 17 if S is a t-secluded star of G with center x, then $N_G(S)$ is a vertex cover of size at most t of F_x. Our kernelization algorithm will first compute, for each $x \in V(G)$, the graph F_x. Then, it performs Buss's kernelization on each graph F_x as described below.

For a vertex $x \in V(G)$, let W_x the set of vertices of degree greater than t in F_x. Note that every vertex cover of size at most t of F_x must contain W_x.

Hence, if $|W_x|$ is greater than t, then x can not be the center of a t-secluded star.

Suppose now that $|W_x| \leq t$ and let $t' = t - |W_x|$. Note that F_x contains a vertex cover of size t if and only if $F_x - W_x$ contains a vertex cover of size at most t'. Since $F_x - W_x$ is a graph of degree at most t, a vertex cover of size t' can cover at most $t \cdot t'$ edges. In other words, if $F_x - W_x$ contains more than $2t \cdot t'$ non isolated vertices, then x can not be the center of a t-secluded star.

Suppose now that $F_x - W_x$ has at most $2t \cdot t'$ non isolated vertices. Let F_x^+ be the subgraph of $G[N_G^2[x]]$, obtained from $F_x - W_x$ removing all the isolated vertices, and then adding x with all its incident edges. Note that F_x^+ contains at most $2t^2 + 1$ vertices. Now let I_x be the set of isolated vertices of $F_x - W_x$ contained in $N_G(x)$. Since $N_G(I_x) \setminus \{x\}$ is contained in W_x, and W_x is contained in $N_G(S)$ for every t-secluded star S with center x, we deduce that I_x may be contained in any such star S. In other words, if S is a star with center x, then S is a solution of Secluded Star on input (G, ω, t, w) if and only if $S - I_x$ is a solution of Secluded Star on input $(F_x^+, \omega^+, t', w')$, where $\omega^+(v) = \omega(v)$ for all $v \in V(F_x^+)$ and $w' = w - \omega(I_x)$.

Now let F_x' be the graph obtained from F_x^+, attaching to each vertex u in $N_G^2(x) \cap V(F_x^+)$ a clique K_u of size $2t$, where all vertices of K_u are adjacent to u. Note that F_x^+ is a graph with at most $4t^3 + 1$ vertices. Moreover, every t'-secluded star in F_x' has center x. Indeed, if the center is some vertex in $N(x)$ or $N_G^2(x)$, then S intersects or has as neighbors more than t vertices in some clique K_u. Let ω' be a function on $V(F_x')$ such that $\omega'(v) = \omega(v)$ if $v \in V(F_x') \cap V(G)$ and $\omega'(v) = 1$ if v is in one of the cliques adjacent to a vertex of $N_G^2(x) \cap V(F_x')$.

We conclude that (F_x', ω', t', w') is a yes-instance of Secluded Star for some $x \in V(G)$ if and only if (G, ω, t, w) is a yes-instance of Secluded Star.

The kernelization algorithm computes for each $x \in V(G)$ the graph F_x and the set W_x. If $W_x \cap N_G^2[x]$ contains more than t vertices the algorithm rejects x and continues with another vertex of $V(G)$. If $|W_x| \leq t$, it computes F_x^+ deleting all the isolated vertices of $F_x - W_x$. If F_x^+ contains more than $2t \cdot t_1 + 1$ vertices, the algorithm rejects x and continues with another vertex of $V(G)$. Finally, the algorithm computes F_x', t', w' and calls the oracle on input (F_x', ω', t', w'). If the oracle answers that (F_x', ω', t', w') is a yes-instance, the algorithm decides that (G, ω, t, w) is a yes-instance. Otherwise the algorithm continues with another vertex of $V(G)$. The algorithm ends when some oracle accepts, or all vertices are rejected. □

It is a natural question to ask whether a polynomial Turing kernelization for Connected Secluded Π-Subgraph is possible for other properties Π.

References

[1] van Bevern, R., Fluschnik, T., Mertzios, G.B., Molter, H., Sorge, M., Suchý, O., 2018. The parameterized complexity of finding secluded solutions to some classical optimization problems on graphs. Discrete Optimization 30, 20–

50. URL: https://doi.org/10.1016/j.disopt.2018.05.002, doi:10.1016/j.disopt.2018.05.002.

[2] van Bevern, R., Fluschnik, T., Tsidulko, O.Y., 2020. Parameterized algorithms and data reduction for the short secluded s-t-path problem. Networks 75, 34–63. URL: https://doi.org/10.1002/net.21904, doi:10.1002/net.21904.

[3] Bodlaender, H.L., Downey, R.G., Fellows, M.R., Hermelin, D., 2009. On problems without polynomial kernels. J. Comput. Syst. Sci. 75, 423–434. doi:10.1016/j.jcss.2009.04.001.

[4] Chechik, S., Johnson, M.P., Parter, M., Peleg, D., 2017. Secluded connectivity problems. Algorithmica 79, 708–741. URL: https://doi.org/10.1007/s00453-016-0222-z, doi:10.1007/s00453-016-0222-z.

[5] Chitnis, R., Cygan, M., Hajiaghayi, M.T., Pilipczuk, M., Pilipczuk, M., 2016. Designing FPT algorithms for cut problems using randomized contractions. SIAM J. Comput. 45, 1171–1229. URL: http://dx.doi.org/10.1137/15M1032077, doi:10.1137/15M1032077.

[6] Cygan, M., Fomin, F.V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S., 2015. Parameterized Algorithms. Springer. URL: http://dx.doi.org/10.1007/978-3-319-21275-3, doi:10.1007/978-3-319-21275-3.

[7] Cygan, M., Lokshtanov, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S., 2019. Minimum bisection is fixed-parameter tractable. SIAM J. Comput. 48, 417–450. URL: https://doi.org/10.1137/140988553, doi:10.1137/140988553.

[8] Downey, R.G., Fellows, M.R., 2013. Fundamentals of Parameterized Complexity. Texts in Computer Science, Springer.

[9] Fomin, F.V., Golovach, P.A., Karpov, N., Kulikov, A.S., 2017. Parameterized complexity of secluded connectivity problems. Theory Comput. Syst. 61, 795–819. URL: https://doi.org/10.1007/s00224-016-9717-x, doi:10.1007/s00224-016-9717-x.

[10] Fomin, F.V., Villanger, Y., 2012. Treewidth computation and extremal combinatorics. Combinatorica 32, 289–308. doi:10.1007/s00493-012-2536-z.

[11] Hüffner, F., Komusiewicz, C., Moser, H., Niedermeier, R., 2009. Isolation concepts for clique enumeration: Comparison and computational experiments. Theor. Comput. Sci. 410, 5384–5397. URL: http://dx.doi.org/10.1016/j.tcs.2009.05.008, doi:10.1016/j.tcs.2009.05.008.

[12] Ito, H., Iwama, K., 2009. Enumeration of isolated cliques and pseudo-cliques. ACM Trans. Algorithms 5, 40:1–40:21. URL: http://doi.acm.org/10.1145/1597036.1597044, doi:10.1145/1597036.1597044.

[13] Johnson, M.P., Liu, O., Rabanca, G., 2014. Secluded path via shortest path, in: Structural Information and Communication Complexity - 21st International Colloquium, SIROCCO 2014, Takayama, Japan, July 23-25, 2014. Proceedings, Springer. pp. 108–120. URL: https://doi.org/10.1007/978-3-319-09620-9_10, doi:10.1007/978-3-319-09620-9_10.

[14] Komusiewicz, C., Hüffner, F., Moser, H., Niedermeier, R., 2009. Isolation concepts for efficiently enumerating dense subgraphs. Theor. Comput. Sci. 410, 3640–3654. URL: http://dx.doi.org/10.1016/j.tcs.2009.04.021, doi:10.1016/j.tcs.2009.04.021.

[15] Lewis, J.M., Yannakakis, M., 1980. The node-deletion problem for hereditary properties is NP-complete. J. Comput. Syst. Sci. 20, 219–230. URL: http://dx.doi.org/10.1016/0022-0000(80)90060-4, doi:10.1016/0022-0000(80)90060-4.

[16] Lokshtanov, D., Ramanujan, M.S., Saurabh, S., Zehavi, M., 2018. Reducing CMSO model checking to highly connected graphs, in: 45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik. pp. 135:1–135:14. URL: https://doi.org/10.4230/LIPIcs.ICALP.2018.135, doi:10.4230/LIPIcs.ICALP.2018.135.

[17] Luckow, M., Fluschnik, T., 2020. On the computational complexity of length- and neighborhood-constrained path problems. Inf. Process. Lett. 156, 105913. URL: https://doi.org/10.1016/j.ipl.2019.105913, doi:10.1016/j.ipl.2019.105913.

[18] Marx, D., 2006. Parameterized graph separation problems. Theor. Comput. Sci. 351, 394–406. URL: http://dx.doi.org/10.1016/j.tcs.2005.10.007, doi:10.1016/j.tcs.2005.10.007.

[19] McConnell, R.M., Spinrad, J.P., 1999. Modular decomposition and transitive orientation. Discrete Mathematics 201, 189–241. URL: https://doi.org/10.1016/S0012-365X(98)00319-7, doi:10.1016/S0012-365X(98)00319-7.

[20] Papadimitriou, C.H., Yannakakis, M., 1996. On limited nondeterminism and the complexity of the V-C dimension. J. Comput. Syst. Sci. 53, 161–170. URL: http://dx.doi.org/10.1006/jcss.1996.0058, doi:10.1006/jcss.1996.0058.

[21] Polishchuk, V., Sedov, L., Voronov, R., 2019. Geometric secluded paths and planar satisfiability. CoRR abs/1902.06471. URL: http://arxiv.org/abs/1902.06471, arXiv:1902.06471.

[22] Yannakakis, M., 1979. The effect of a connectivity requirement on the complexity of maximum subgraph problems. J. ACM 26, 618–630. URL: http://doi.acm.org/10.1145/322154.322157, doi:10.1145/322154.322157.