Structure Theorem and Isomorphism Test for Graphs with Excluded Topological Subgraphs

Martin Grohe
Institut für Informatik
Humboldt-Universität zu Berlin
Berlin, Germany
grohe@informatik.hu-berlin.de

Dániel Marx*
Computer and Automation Research Institute,
Hungarian Academy of Sciences (MTA SZTAKI)
Budapest, Hungary
dmarx@informatik.hu-berlin.de

ABSTRACT

We generalize the structure theorem of Robertson and Seymour for graphs excluding a fixed graph H as a minor to graphs excluding H as a topological subgraph. We prove that for a fixed H, every graph excluding H as a topological subgraph has a tree decomposition where each part is either "almost embeddable" to a fixed surface or has bounded degree with the exception of a bounded number of vertices. Furthermore, such a decomposition is computable by an algorithm that is fixed-parameter tractable with parameter $|H|$.

We present two algorithmic applications of our structure theorem. To illustrate the mechanics of a "typical" application of the structure theorem, we show that on graphs excluding H as a topological subgraph, Partial Dominating Set (find k vertices whose closed neighborhood has maximum size) can be solved in time $f(H,k) \cdot n^{O(1)}$ time. More significantly, we show that on graphs excluding H as a topological subgraph, Graph Isomorphism can be solved in time $n^{f(H)}$. This result unifies and generalizes two previously known important polynomial-time solvable cases of Graph Isomorphism: bounded-degree graphs [18] and H-minor free graphs [22]. The proof of this result needs a generalization of our structure theorem to the context of invariant treelike decomposition.

Categories and Subject Descriptors

F.2 [**Theory of Computing**]: Analysis of Algorithms and Problem Complexity; G.2.2 [**Mathematics of Computing**]: Discrete Mathematics—*Graph Theory*

General Terms

Algorithms

*Research supported by the European Research Council (ERC) grant 280152.

STOC'12, May 19–22, 2012, New York, New York, USA.

Keywords

topological minors, fixed-parameter tractability, graph isomorphism

1. INTRODUCTION

We say that a graph H is a *minor* of G if H can be obtained from G by deleting vertices, deleting edges, and contracting edges. A graph G is *H-minor free* if H is not a minor of G. Robertson and Seymour [25] proved a structure theorem for the class of H-minor graphs: roughly speaking, every H-minor free graph can be decomposed in a way such that each part is "almost embeddable" into a fixed surface. This structure theorem has important algorithmic consequences: many natural computational problems become easier when restricted to H-minor free graphs [4, 13, 6, 15, 14, 5, 10]. These algorithmic results can be thought of as far-reaching generalizations of algorithms on planar graphs and bounded-genus surfaces.

A more general way of defining restricted classes of graphs is to exclude topological subgraphs instead of minors. A graph H is a *topological subgraph* (or *topological minor*) of graph G if a subdivision of H is a subgraph of G. It is easy to see that if H is a topological subgraph of G, then H is also a minor of G. Thus the class of graphs excluding H as a topological subgraph is a more general class than H-minor free graphs.

One can ask if graphs excluding H as a topological subgraph admit a similar structure theorem as H-minor free graphs. However, graphs excluding a topological subgraph can be much more general. For example, no 3-regular graph can contain a subdivision of K_5 (as K_5 is 4-regular). Therefore, the class of graphs excluding K_5 as a topological subgraph includes in particular every 3-regular graph. This suggests that it is unlikely that this class can be also characterized by (almost) embeddability into surfaces.

Nevertheless, our first result is a structure theorem for graphs excluding a graph H as a topological subgraph. We prove that, in some sense, only the bounded-degree graphs make this class more general than H-minor free graphs. More precisely, we prove a structure theorem that decomposes graphs excluding H as a topological subgraph into almost bounded-degree parts and into H'-minor free parts (for some other graph H'). The H'-minor free parts can be further refined into almost-embeddable parts using the structure theorem of Robertson and Seymour [25], to obtain our main structural result (see Corollary 4.4 for the precise statement):

THEOREM 1.1 (INFORMAL). *For every fixed graph H, every graph excluding H as a topological subgraph has a tree decomposition where every torso*

(i) either has bounded degree with the exception of a bounded number of vertices, or
(ii) almost embeddable into a surface of bounded genus.

Furthermore, such a decomposition can be computed in time $f(H) \cdot |V(G)|^{O(1)}$ for some computable function f.

Our structure theorem allows us to lift problems that are tractable on both bounded-degree graphs and on H-minor free graphs to the class of graphs excluding H as a topological subgraph. We demonstrate this principle on the PARTIAL DOMINATING SET problem (find k vertices whose closed neighborhood is maximum). Following a bottom-up dynamic programming approach, we solve the problem in each bag of the tree decomposition (using the fact that the problem can be solved in linear-time on both bounded-degree and on almost-embeddable graphs).

THEOREM 1.2. PARTIAL DOMINATING SET *can be solved in time $f(k,H) \cdot n^{O(1)}$ when restricted to graphs excluding H as a topological subgraph.*

One could prove similar results for other basic problems such as INDEPENDENT SET or DOMINATING SET. However, a result of Dvorak et al. [7] shows that problems expressible in first-order logic can be solved in linear time on classes of graphs having bounded expansion, and therefore on graphs excluding H as a topological subgraph. The problems INDEPENDENT SET and DOMINATING SET (for a fixed k) can be expressed in first-order logic, thus the analogs of Theorem 1.2 for these problems follow from [7]. On the other hand, PARTIAL DOMINATING SET is not expressible in first-order logic, hence the techniques of Dvorak et al. [7] do not apply to this problem.

The main algorithmic result of the paper concerns the GRAPH ISOMORPHISM problem (given graphs G_1 and G_2, decide if they are isomorphic). GRAPH ISOMORPHISM is known to be polynomial-time solvable for bounded-degree graphs [18, 2] and for H-minor free graphs [22, 9]. In fact, for these classes of graphs, even the more general canonization problem can be solved in polynomial time: there is an algorithm labeling the vertices of the graph with positive integers such that isomorphic graphs get isomorphic labelings. It is tempting to expect that our structure theorem together with a bottom-up strategy give a canonization algorithm for graphs excluding H as a topological subgraph: in each bag, we use the canonization algorithm either for bounded-degree graphs or H-minor free graphs (after encoding somehow the canonized versions of the child bags, which seems to be a technical problem only). However, this approach is inherently doomed to failure: there is no guarantee that our decomposition algorithm produces isomorphic decompositions for isomorphic graphs. Therefore, even if two graphs are isomorphic, the bottom-up canonization algorithm could be working on two completely different decompositions and therefore could obtain different results on the two graphs.

We overcome this difficulty by generalizing our structure theorem to the context of treelike decompositions introduced by the first author in [11, 9]. A treelike decomposition is similar to a tree decomposition, but it is defined over a directed acyclic graph instead of a rooted tree, and therefore it contains several tree decompositions. The Invariant Decomposition Theorem (Section 8) generalizes the structure theorem by giving an algorithm that computes a treelike decomposition in a way that the decompositions obtained for isomorphic graphs are isomorphic. Then the Lifting Lemma (Section 9) formalizes the bottom-up strategy informally described in the previous paragraph: if we can compute treelike decompositions for a class of graphs in an invariant way and we have a canonization algorithm for the bags, then we have a canonization algorithm for this class of graphs. Although the idea is simple, in order to encode the child bags, we have to state this algorithmic result in a more general form: instead of graphs, we have to work with weighted relational structures. This makes the statement and proof of the Lifting Lemma more technical. Putting together these results, we obtain:

THEOREM 1.3. *For every fixed graph H,* GRAPH ISOMORPHISM *can be solved in polynomial-time restricted to graphs excluding H as a topological subgraph.*

Let us quickly remark that it is unlikely that Theorem 1.3 could be generalized to all classes of graphs with bounded expansion, as the isomorphism problem on such a class can be as hard as on general graphs. To see this, consider two graphs G_1 and G_2 on n-vertices and let us obtain G_1' and G_2' by subdividing each edge with n new vertices. Now G_1' and G_2' have bounded expansion and they are isomorphic if and only if G_1 and G_2 are.

Actually, we not only obtain a polynomial time isomorphism test, but also a polynomial time canonisation algorithm. Our theorem generalizes and unifies the results of Babai and Luks [18, 2] on bounded-degree graphs and of Ponomarenko [22] on H-minor free graphs. Let us remark that Ponomarenko's result implies that there is a polynomial time isomorphism test for all classes of graphs of bounded genus, which has been proved earlier by Filotti and Mayer [8] and Miller [21], and for all classes of graphs of bounded tree width, which was also proved later (independently) by Bodlaender [3]. Miller [20] gave a common generalization of the bounded degree and bounded genus classes to classes that he called k-contractible. These classes do not seem to have a simple graph-theoretic characterization; they are defined in terms of properties of the automorphism groups needed for the algorithm. Excluding topological subgraphs, on the other hand, is a natural graph theoretic restriction that generalizes both bounding the degree and excluding minors and hence bounding the genus.

For the convenience of the reader, let us summarize how the different results in the present paper depend on previous results in the literature:

- The proof of the existence of the decomposition into H-minor free and almost bounded-degree parts is self-contained. The algorithm computing such a decomposition needs the minor testing algorithm of [24] or [16].
- The proof of the existence of the more refined decomposition into almost-embeddable and almost bounded-degree parts needs the graph structure theorem of Robertson and Seymour [25]. The algorithm computing such a decomposition needs the algorithmic version of the structure theorem [5]; to achieve $f(H) \cdot n^{O(1)}$ running time, a more recent stronger algorithmic result is needed [17].

- The algorithm for Partial Dominating Set needs the more refined decomposition, hence it relies on [24, 17]. Additionally, it needs the fact proved in [10] that almost-embeddable graphs have bounded local treewidth.
- The result on Graph Isomorphism needs the minor testing algorithm of [24] or [16] to compute the treelike decomposition. Additionally, the canonization algorithms for bounded-degree graphs [2] and for H-minor free graphs ([22] or [9]) are needed.

Note that none of the results rely on the topological subgraph testing algorithm of [12] or need any substantial result from the monograph [9].

The paper is organized as follows. Sections 2–3 introduce the notation used in the paper. Section 4 states the structure theorem and shows how it can be proved by appropriate local decomposition lemmas. Section 5 introduces the notion of tangles, which is an important tool in the proofs of the local decomposition lemmas in Section 6. Section 7 uses the structure theorem in an algorithm for Partial Dominating Set. Section 8 introduces treelike decomposition and proves the Invariant Decomposition Theorem. Section 9 proves the Lifting Lemma for canonizations, completing the proof of Theorem 1.3.

2. PRELIMINARIES

$\mathbb{Z}$ and $\mathbb{N}$ denote the sets of integers and nonnegative integers, respectively. For $m, n \in \mathbb{Z}$, we let $[m,n] := \{\ell \in \mathbb{Z} \mid m \le \ell \le n\}$ and $[n] := [1,n]$. The power set of a set S is denoted by 2^S, and the set of all k-element subsets of S by $\binom{S}{k}$. For a mapping f defined on S, we let $f(S) := \{f(s) \mid s \in S\}$. The cardinality of a set S is denoted by $|S|$.

Let G be a graph. The *order* of a graph G is $|G| := |V(G)|$. The set of all neighbors of a vertex $v \in V(G)$, called the *open neighborhood of* v, is denoted by $N^G(v)$. The *closed neighborhood* of v is the set $N^G[v] := \{v\} \cup N^G(v)$. The *closed* and *open neighborhood* of a subset $W \subseteq V(G)$ are the sets $N^G[W] := \bigcup_{w \in W} N^G[w]$ and $N^G(W) := N^G[W] \setminus W$, respectively, and the *closed* and *open neighborhood* of a subgraph $H \subseteq G$ are the sets $N^G[H] := N^G[V(H)]$ and $N^G(H) := N^G(V(H))$, respectively. We omit the index ^{G} if G is clear from the context, and we do the same for similar notations introduced later. We let $\partial^G(W) = |N^G(W)|$.

For every set V, we let $K[V]$ be the complete graph with vertex set V, and for every $n \in \mathbb{N}$, we let $K_n := K\big[[n]\big]$.

Let G be a graph. A graph H is a *minor* of G (denoted by $H \preceq G$) if H can be obtained from G by deleting vertices, deleting edges, and contracting edges. Equivalently, we can define $H \preceq G$ the following way. Two sets $S, T \subseteq V(G)$ *touch* if either $S \cap T = \varnothing$ or there is an edge $vw \in V(G)$ such that $v \in S$ and $w \in T$. It can be shown that $H \preceq G$ if and only if there is a family $(I_w)_{w \in V(H)}$ of pairwise disjoint connected subsets of $V(G)$ such that for every $u, v \in V(H)$ that are adjacent in H, the sets I_u and I_v touch in G. We call this family I an *image* of H in G and the sets I_w are the *branch sets* of the image.

Theorem 2.1 ([24, 16]). *There is an $f(H) \cdot |V(G)|^3$ time algorithm (for some computable f) that finds a H-minor image in G, if exists.*

A *subdivision* H' of a graph H is obtained by replacing each edge of H by a path of length at least 1. We say that H is a *topological subgraph* (or *topological minor* of G) and denote it by $H \preceq_T G$ if a subdivision of H is a subgraph of G. Equivalently, H is a topological subgraph of G if H can be obtained from G by deleting edges, deleting vertices, and dissolving degree 2 vertices (which means deleting the vertex and making its two neighbors adjacent). For fixed H, it can be decided in cubic time whether a graph G contains a subdivision of H (although we do not need this result in the current paper):

Theorem 2.2 ([12]). *There is an $f(H) \cdot |V(G)|^3$ time algorithm (for some computable f) that finds a subdivision of H in G, if exists.*

Let D be digraph. For every $t \in V(D)$, we let $N^D(t) := \{u \in V(D) \mid tu \in E(D)\}$. We call vertices of in-degree 0 *roots* and vertices of out-degree 0 *leaves* of D. The *height* of an acyclic digraph D is the length of the longest path in D.

It will be convenient for us to view trees as being directed, unless we explicitly call them undirected. Hence for us, a *tree* is an acyclic digraph T that has a unique node (the *root*) such that for every node t there is a exactly one path from $r(T)$ to t.

For two graphs A and B, the graph $A \cup B$ is defined by $V(A \cup B) = V(A) \cup V(B)$ and $E(A \cup B) = E(A) \cup E(B)$. Let G be a graph. A *separation* of G is a pair (A,B) of subgraphs of G such that $A \cup B = G$ and $E(A \cap B) = \varnothing$. The *order* of a separation (A,B) is $|V(A) \cap V(B)|$.

3. TREE DECOMPOSITIONS

A *tree decomposition* of a graph is a pair (T, β), where T is a rooted tree and $\beta : V(T) \to 2^{V(G)}$, such that for all nodes $v \in V(G)$ the set $\{t \in V(G) \mid v \in \beta(t)\}$ is nonempty and connected in the undirected tree underlying T, and for all edges $e \in E(G)$ there is a $t \in V(T)$ such that $e \subseteq \beta(t)$. Most readers will be familiar with this definition, but it will be convenient for us to view tree decompositions from a different perspective here.

If (T, β) is a tree decomposition of a graph G, then we define mappings $\sigma, \gamma, \alpha : V(T) \to 2^{V(G)}$ by letting for all $t \in V(T)$

$$\sigma(t) := \begin{cases} \varnothing & \text{if } t \text{ is the root of } T, \\ \beta(t) \cap \beta(s) & \text{if } s \text{ is the parent of } t \text{ in } T, \end{cases} \tag{3.1}$$

$$\gamma(t) := \bigcup_{u \text{ is a descendant of } t} \beta(u), \tag{3.2}$$

$$\alpha(t) := \gamma(t) \setminus \sigma(t). \tag{3.3}$$

We call $\beta(t), \sigma(t), \gamma(t), \alpha(t)$ the *bag* at t, *separator* at t, *cone* at t, *component* at t, respectively. It is easy to verify that the following conditions hold:

(TD.1) T is a tree.

(TD.2) For all $t \in V(D)$ it holds that $\alpha(t) \cap \sigma(t) = \varnothing$ and $N^G(\alpha(t)) \subseteq \sigma(t)$.

(TD.3) For all $t \in V(D)$ and $u \in N^D(t)$ it holds that $\alpha(u) \subseteq \alpha(t)$ and $\gamma(u) \subseteq \gamma(t)$.

(TD.4) For all $t \in V(D)$ and all distinct $u_1, u_2 \in N^D(t)$ it holds that $\gamma(u_1) \cap \gamma(u_2) = \sigma(u_1) \cap \sigma(u_2)$.

(TD.5) For the root r of T it holds that $\sigma(r) = \varnothing$ and $\alpha(r) = V(G)$.

Conversely, consider a triple (T, σ, α), where T is a digraph and $\sigma, \alpha : V(T) \to 2^{V(G)}$. We define $\gamma, \beta : V(T) \to 2^{V(G)}$ by

$$\gamma(t) := \sigma(t) \cup \alpha(t), \tag{3.4}$$

$$\beta(t) := \gamma(t) \setminus \bigcup_{u \in N^T(t)} \alpha(u) \tag{3.5}$$

for all $t \in V(T)$. Then it is easy to prove that if (TD.1)–(TD.5) are satisfied, then (T, β) is a tree decomposition (see [9] for a proof). Thus we may also view triples (T, σ, α) satisfying (TD.1)–(TD.5) as tree decompositions. We jump back and forth between both versions of tree decompositions, whichever is more convenient. The treelike decompositions introduced in Section 8 need to be defined as triples (T, σ, α), thus looking at tree decompositions also this way in the first part of the paper makes the transition between the two concepts smoother.

Let (T, β) be a tree decomposition of a graph G. The *width* of (T, β) is $\max\{|\beta(t)|-1 \mid t \in V(T)\}$, and the *adhesion* of (T, β) is $\max\{|\sigma(t)| \mid t \in V(T)\}$. The *tree width* of a graph is the minimum possible width of a tree decomposition of G. However, in the current paper, rather than minimizing tree width (i.e., minimizing the size of the bags), we are mostly interested in decompositions where the graph induced by each bag (plus some additional edges) is "nice" in a certain sense. For every node $t \in V(T)$, the *torso* at t is the graph

$$\tau(t) := G[\beta(t)] \cup K[\sigma(t)] \cup \bigcup_{u \in N^D(t)} K[\sigma(u)]. \tag{3.6}$$

That is, we take the graph induced by bag $\beta(t)$, turn $\sigma(t)$ into a clique, and make vertices x, y adjacent if they appear together in the separator (or equivalently, the cone) of some child u of t. For a class $\mathcal{A}$ of graphs, (T, β) is a tree decomposition *over* $\mathcal{A}$ if all its torsos are in $\mathcal{A}$.

A related notion is the *torso* of G with respect to a set $C \subseteq V(G)$, denoted by $\mathrm{torso}(G, C)$, which is defined as graph on C where $u, v \in V(G)$ are adjacent if there is a path P in G with endpoints u and v such that the internal vertices of P are disjoint from C. In other words,

$$\mathrm{torso}(G, C) := G[C] \cup \bigcup_{X \text{ is a component of } G \setminus C} K[N^G(X)].$$

It is easy to see that $\tau(G, \beta(t)) \subseteq \tau(t)$. Equality is not true in general: $G[\alpha(u)]$ for some $u \in N^D(t)$ is not necessarily connected, thus it is not necessarily true that $\sigma(u)$ is $N^G(X)$ for some component X of $G \setminus \beta(t)$.

4. LOCAL AND GLOBAL STRUCTURE THEOREMS

The main structural result of the paper is a decomposition theorem for graphs excluding a topological subgraph:

THEOREM 4.1 (GLOBAL STRUCTURE THEOREM). *For every $k \in \mathbb{N}$, there exists constants $a(k)$, $b(k)$, $c(k)$, $d(k)$, $e(k)$, such that the following holds. Let H be a graph on k vertices. Then for every graph G with $H \not\preceq_T G$ there is a tree decomposition (T, β) of adhesion at most $a(k)$ such that for all $t \in V(T)$ one of the following three conditions is satisfied:*

(i) $|\beta(t)| \leq b(k)$.
(ii) $\tau(t)$ *has at most* $c(k)$ *vertices of degree larger than* $d(k)$.
(iii) $K_{e(k)} \not\preceq \tau(t)$.

Furthermore, there is an algorithm that, given graphs G, H of sizes n, k, respectively, in time $f(k) \cdot n^{O(1)}$ for some computable function f, computes either such a decomposition (T, β) or a subdivision of H in G.

The reader could find it convenient to refer to the constants a, b, c, d, e as the bounds on the adhesion, bag size, number of apices, maximum degree, and excluded clique. We remark that all the constants are polynomially large. Note that (i) is redundant: by choosing $d(k)$ or $e(k)$ sufficiently large, a bag satisfying (i) trivially satisfies (ii) and (iii). We state the result this way, because it shows the high-level structure of the proof, which involves three decomposition results corresponding to the three cases.

The proof of the Global Structure Theorem 4.1 builds a tree decomposition step by step, iteratively decomposing the graph locally in each step. The Local Structure Theorem describes the "local" structure of a graph, as seen from a single node of a tree decomposition. We describe this local structure in terms of star decompositions, to be defined next. A *star* is a tree of height 1. We usually call the root of a star its *center* and the leaves of a star its *tips*. A *star decomposition* of a graph G is a tree decomposition (T, β) where T is a star. Note that if (T, β) is a star decomposition, then for every tip t of the star T it holds that $\beta(t) = \gamma(t)$.

THEOREM 4.2 (LOCAL STRUCTURE THEOREM). *For every $k \in \mathbb{N}$, there exists constants $a(k)$, $b(k)$, $c(k)$, $d(k)$, $e(k)$ such that the following holds. There is an $f(k) \cdot |V(G)|^{O(1)}$ time algorithm that, given a graph G, a set S of size $\leq a(k)$, and an integer k,*

(1) either returns a subdivision of K_k in G,
(2) or computes a star decomposition $\Sigma_S = (T_S, \alpha_S, \sigma_S)$ of $G \cup K[S]$ of adhesion $\leq a(k)$ such that $S \subseteq \beta_S(s)$ for the center s, $\alpha(t) \subset \alpha(s)$ for every tip t, and one of the following three conditions is satisfied:
  (a) $|\beta_S(s)| \leq b(k)$.
  (b) $\tau_S(s)$ *does not contain a* $K_{e(k)}$*-minor.*
  (c) At most $c(k)$ *vertices of* $\tau_S(s)$ *have degree more than* $d(k)$ *in* $\tau(s)$.

The condition that $\alpha(t)$ is a proper subset of $\alpha(s)$ makes sure that we make progress and compute a tree decomposition after a finite number of applications of Theorem 4.2. Note the technical detail that Σ_S in (2) is a decomposition of $G \cup K[S]$ instead of G. As $G \cup K[S]$ has more edges than G, this makes the statement slightly stronger (because it makes harder to satisfy the requirements on $\tau_S(s)$). The proof of the Global Structure Theorem 4.1 needs this extra condition, since the set S will connect the graph to the part of the tree decomposition already computed. In (1), however, the K_k-subdivision is found in G (which is a slightly stronger statement than finding it in $G \cup K[S]$).

The proof of the Global Structure Theorem 4.1 follows from the Local Structure Theorem by a fairly simple induction (see below). In Section 4.2, we show that Local Structure Theorem 4.2 can be proved by putting together three decomposition lemmas. We prove these lemmas in Sections 5–6. Let us remark the Global Structure Theorem can be seen as an instance of a general theorem due to Robertson and Seymour [23, (11.3)], explaining how to construct a tree decomposition whose torsos have a "nice structure" in graphs with a "nice local structure", where the local structure is described with respect to a tangle (see Section 5). Our proof

follows the ideas of Robertson and Seymour's construction, but as Robertson and Seymour's theorem is not algorithmic, and since there would be a large notational overhead, we see no benefit in appealing to Robertson and Seymour's theorem here and instead carry out our own version of the construction, which is not very difficult anyway.

PROOF (OF THE GLOBAL STRUCTURE THEOREM 4.1). Let $a(k)$, $b(k)$, $c(k)$, $d(k)$, $e(k)$ as in the Local Structure Theorem 4.2. Let G be a graph. We shall describe the construction of a tree decomposition (T, β) of G satisfying all conditions asserted in the lemma. The construction may fail, but in that case it yields a subdivision of H in G.

We will built the tree T inductively starting from the root. For every node t we will define the set $N^T(t)$ of its children and sets $\sigma(t), \alpha(t)$ such that $|\sigma(t)| \leq a(k)$ and $N^G(\alpha(t)) \subseteq \sigma(t)$. As usual, we define $\gamma(t)$, $\beta(t)$, and $\tau(t)$ as in (3.4), (3.5), and (3.6). In each step, we will prove that $\tau(t)$ satisfies one of (i), (ii), or (iii).

We start with a root r of T and let $\sigma(r) := \varnothing$ and $\alpha(r) := V(G)$. For the inductive step, let t be a node for which $\sigma(t)$ and $\alpha(t)$ are defined, but $N^D(t)$ is not yet defined. We let $G_t := G[\gamma(t)]$. Let us run the algorithm of Theorem 4.2 on G_t (as G), $\sigma(t)$ (as S), and k. If it returns a subdivision of K_k in G_t, then we can clearly return a subdivision of H in G and we are done. Otherwise, it returns a star decomposition $\Sigma_t := (T_t, \sigma_t, \alpha_t)$ of $G \cup K[\sigma(t)]$ having adhesion at most $a(k)$; let s_t be the center of T_t. We let $N^T(t) := V(T_t) \setminus \{s_t\}$ be the set of tips of T_t, where without loss of generality we assume that this set is disjoint from the tree T constructed so far. For every $u \in N^T(t)$ we let $\sigma(u) := \sigma_t(u)$ and $\alpha(u) := \alpha_t(u)$. Observe that we have $\beta(t) = \gamma(t) \setminus \bigcup_{u \in N^T(t)} \alpha(u) = \beta_t(s_t)$. Furthermore, since Σ_t is a decomposition of $G \cup K[\sigma(t)]$ and $\sigma(t)$ induces a clique in $G \cup K[\sigma(t)]$, we have that $\tau(t) = \tau_t(s_t)$. Thus one of the three cases of Theorem 4.2 holds for the bag $\beta(t)$ as well.

To see that (T, β) is a tree decomposition, it is easiest to verify it satisfies (TD.2)–(TD.4): it follows from the fact that the star decomposition Σ_t used in each step of the construction does satisfy these conditions. Condition (TD.1) is obvious and (TD.5) follows because we start the construction with a node t having $\alpha(t) = V(G)$ and $\sigma(t) = \varnothing$. Note that the bound $a(k)$ on the adhesion of Σ_t implies the same bound on the adhesion of (T, β).

To see that the construction terminates, note that for all $t \in V(T)$, Theorem 4.2 states that $\alpha_t(u) \subset \alpha_t(s_t)$ for every tip u of T_t. This means that that $\alpha(u) \subset \alpha(t)$ holds for every $u \in N^T(t)$ and hence the height of the tree is at most $|V(G)|$. Moreover, $\alpha(u_1)$ and $\alpha(u_2)$ are disjoint for two distinct children of node t and it follows that the total number of leaves can be bounded by $|V(G)|$. Thus the algorithm, excluding the calls to Theorem 4.2, runs in polynomial time. The claim on the running time follows from Theorem 4.2. □

4.1 Almost Embeddable Graphs and a Refined Structure Theorem

In this section, we combine our structure theorem with Robertson and Seymour's structure theorem for graphs with excluded minors [25], which says that for graph H, all graphs excluding H as a minor have a tree decomposition into torsos that are almost embeddable into some surface.

We start by reviewing Robertson and Seymour's structure theorem. We need first the definition of (p, q, r, s)-almost embeddable graphs (for the current paper, the exact definition will not be important, thus the reader can safely skip the details). We assume that the reader is familiar with the basics of surface topology and graph embeddings. A *path decomposition* is a tree decomposition (P, β) where P is a path. For every $n \in \mathbb{N}$, by P^n we denote the path with vertex set $[n]$ and edges $i(i+1)$ for all $i \in [n-1]$. A *p-ring* is a tuple $(R, v_1, \ldots, v_n)$, where R is a graph and $v_1, \ldots, v_n \in V(R)$ such that there is a path decomposition (P^n, β) of R of width p with $v_i \in \beta(i)$ for all $i \in [n]$. A graph G is *(p, q)-almost embedded* in a surface $\mathbf{S}$ if there are graphs $G_0, G_1, \ldots, G_q$ and mutually disjoint closed disks $\mathbf{D}_1, \ldots, \mathbf{D}_q \subseteq \mathbf{S}$ such that:

(i) $G = \bigcup_{i=0}^{q} G_i$.
(ii) G_0 is embedded in $\mathbf{S}$ and has a nonempty intersection with the interiors of the disks $\mathbf{D}_1, \ldots, \mathbf{D}_q$.
(iii) The graphs $G_1, \ldots, G_q$ are mutually disjoint.
(iv) For all $i \in [q]$ we have $E(G_0 \cap G_i) = \varnothing$, and there are $n_i \in \mathbb{N}$ and $v_1^i, \ldots, v_{n_i}^i \in V(G)$ such that $V(G_0 \cap G_i) = \{v_1^i, \ldots, v_{n_i}^i\}$, and the vertices $v_1^i, \ldots, v_{n_i}^i$ appear in cyclic order on the boundary of the disk $\mathbf{D}_i$.
(v) For all $i \in [q]$ the tuple $(G_i, v_1^i, \ldots, v_{n_i}^i)$ is a p-ring.

A graph G is *(p, q, r, s)-almost embeddable* if there is an *apex set* $X \subseteq V(G)$ of size $|X| \leq s$ such that $G \setminus X$ is isomorphic to a graph that is (p, q)-almost embedded in a surface of Euler genus r.

THEOREM 4.3 ([25, 17]). *For every graph H there are constants $p, q, r, s \in \mathbb{N}$ such that every graph G with $H \not\preceq G$ has a tree decomposition (T, β) such that for all $t \in V(T)$ the torso $\tau(t)$ is (p, q, r, s)-almost embeddable.*

Furthermore, there is an algorithm that, given G and H, in time $f(|H|) \cdot n^3$ for some computable function f, either finds a H-minor image in G, or computes such a tree decomposition and moreover, computes an apex set Z_t of size at most s for every $t \in V(T)$.

As a corollary of this theorem and our structure theorem we get:

COROLLARY 4.4. *For every graph H there are constants $c, d, p, q, r, s \in \mathbb{N}$ such that every graph G with $H \not\preceq_T G$ has a tree decomposition (T, β) such that for all $t \in V(T)$,*

(i) either $\tau(t)$ is (p, q, r, s)-almost embeddable,
(ii) or at most c vertices of $\tau(t)$ have degree greater than d.

Furthermore, there is an algorithm that, given G and H, in time $f(|H|) \cdot n^{O(1)}$ for some computable function f, either finds a subdivision of H in G, or computes such a tree decomposition, and moreover computes an apex set Z_t of size at most s for every bag of the first type.

4.2 The Three Local Decomposition Lemmas

We prove the Local Structure Theorem 4.2 by stacking three decomposition lemmas on top of each other (see Figure 4.1). Each lemma provides either a star decomposition corresponding to one of the three cases (i)–(iii) or an "obstruction" which can be feeded into the next lemma as input.

The first decomposition lemma either finds a star decomposition where the center bag has bounded size or finds a "highly connected" set in the following sense:

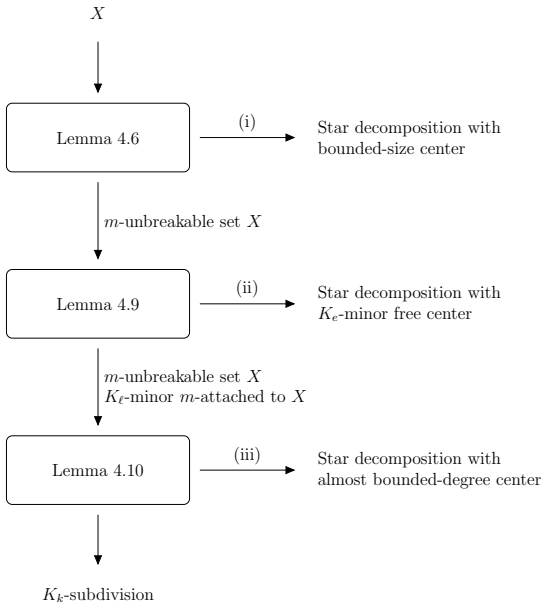

Figure 4.1: The three decomposition lemmas in the proof of Local Structure Theorem 4.2.

Definition 4.5. *Let G be a graph and $X \subseteq V(G)$. A separation (A,B) of G* breaks *X if $|(V(A)\cap X)\cup V(A\cap B)| < |X|$ and $|(V(B)\cap X)\cup V(A\cap B)| < |X|$.*

*The set X is m-*unbreakable *if there is no separation (A,B) of G of order $< m$ that breaks X.*

There is a simple way of detecting if a set X is m-unbreakable by considering all possible ways of breaking X. Note that the running time of the following algorithm is exponential in the size of the set, but we will use it only on unbreakable sets of bounded size.

Lemma 4.6. *There is an algorithm that, given a graph G and a set $X \subseteq V(G)$ and a $m \in \mathbb{N}$, either computes a separation of G of order $< m$ that breaks X or correctly decides that $|X|$ is m-unbreakable in time $3^{|X|}n^{O(1)}$.*

It is not difficult to see that a large unbreakable set is an obstruction for having small treewidth, that is, for having a tree decomposition where every bag has small size. Therefore, it is not surprising that the proof of the first local decomposition lemma is very similar to algorithms finding tree decompositions.

Lemma 4.7 (Bounded-size star decomposition). *For every $m \in \mathbb{N}$, there is a constant $b^*(m)$ such that the following holds. There is an $f(m)\cdot|V(G)|^{O(1)}$ time algorithm that, given a graph G, an integer m, a set X of size $\leq 3m-2$, and an integer k,*

(1) either finds an m-unbreakable set $X' \supseteq X$ of size $3m-2$.
(2) or computes a star decomposition $\Sigma_X = (T_X, \alpha_X, \sigma_X)$ of $G \cup K[X]$ having adhesion $< 3m-2$ such that $X \subseteq \beta_X(s)$ and $|\beta_X(s)| \leq b^*(m)$ for the center s of T_X.

Proof. Let $b^*(m) = 4m-3$. If $|V(G)| < 3m-2$, then we can return a star decomposition consisting of a single center node s with $\alpha(s) = V(G)$ and $\sigma(s) = \varnothing$. Otherwise, let X' be an arbitrary superset of X having size $3m-2$. Let us use the algorithm of Lemma 4.6 to test if X' is m-unbreakable; if so, then we can return X' and we are done. Otherwise, there is a separation (A,B) of G having order $< m$ such that $|(X'\cap V(A))\cup Q|, |(X'\cap V(B))\cup Q| < |X'| = 3m-2$ for $Q = V(A)\cap V(B)$. Let us construct a star decomposition $\Sigma_X = (T_X, \alpha_X, \sigma_X)$ with center s and a tips t_A, t_B. First, let $\alpha(s) = V(G)$ and $\sigma(s) = \varnothing$. Let $\alpha(t_A) = V(A)\setminus(Q\cup X')$ and $\sigma(t_A) = |(X'\cap V(A))\cup Q|$; it is clear that $|\sigma(t_A)| < 3m-2$. Similarly, let $\alpha(t_B) = V(B)\setminus(Q\cup X')$ and $\sigma(t_B) = |(X'\cap V(B))\cup Q|$. It is straightforward to verify that this is indeed a star decomposition of $G\cup K[X]$ with adhesion $< 3m-2$. Furthermore, $|\beta(s)| = |Q\cup X'| \leq m-1+3m-2 = b^*(m)$. □

The second local decomposition lemma takes an unbreakable set X of appropriate size, and either finds a star decomposition where the torso of the center node excludes some minor or finds a large clique minor. Furthermore, this clique minor has the additional property that it is close to the unbreakable set X in the following sense:

Definition 4.8. *Let I be an H-minor image in G and let X be a set of vertices. We say that I is m-*attached *to X if there is no separation (A,B) of order $< m$ such that $I(v) \subseteq V(A)\setminus V(B)$ for some $v \in V(H)$ and $|(V(B)\cap X)\cup V(A\cap B)| \geq |X|$.*

In particular, if X is an m-unbreakable set and I is m-attached to X, then whenever $I(v) \subseteq V(A)\setminus V(B)$ for some $v \in V(H)$ and separation (A,B) of order $< m$, then we know that $|(V(A)\cap X)\cup V(A\cap B)| \geq |X|$. Thus in every every separation, I is on the same side as the larger part of X.

Lemma 4.9 (Excluded-minor star decomposition). *For every $\ell, m \in \mathbb{N}$, there is a constant $e^*(\ell,m)$ such that the following holds. There is an $f(\ell,m)\cdot|V(G)|^{O(1)}$ time algorithm that, given a graph G, integers ℓ, m, and an m-unbreakable set X of size $3m-2$*

(1) either finds a K_ℓ-minor image I in G that is m-attached to X,
(2) or computes a star decomposition $\Sigma_X = (T_X, \alpha_X, \sigma_X)$ of $G\cup K[X]$ having adhesion $< |X|$ such that $X \subseteq \beta_X(s)$ and $\tau_X(s)$ does not contain a $K_{e^*(\ell,m)}$-minor for the center s of T_X.

Furthermore, suppose that the algorithm computes Σ_X on input (G,X) and let (G',X') be a pair such that there is an isomorphism f from G to G' with $f(X) = X'$. Then the algorithm computes a star decomposition $\Sigma'_{X'}$ on input (G',X') and there exists an isomorphism g from T_X to $T_{X'}$ such that for all $t \in V(T_X)$ we have $\sigma_{X'}(g(t)) = f(\sigma_X(t))$ and $\alpha_{X'}(g(t)) = f(\alpha_X(t))$.

Lemma 4.9 states an invariance condition saying that for isomorphic input the decomposition is isomorphic. This condition is not required for the proof of the Global Structure Theorem 4.1, but will be essential for the proof of the Invariant Decomposition Theorem 8.3 in Section 8. Note that Lemma 4.7 does not state such an invariance condition and in fact there does not seem to be an obvious way of ensuring invariance (for example, already the selection of X' in the first step of the proof is completely arbitrary and hence cannot be done in an invariant way). This is precisely the reason why we need to use the more general treelike decompositions in Sections 8–9 if we want the construction to be invariant.

The proof of Lemma 4.9 is deferred to Section 6.1. The algorithm repeatedly finds K_ℓ-minor images and tests if they are m-attached to S. If so, it returns it, otherwise there is a separator that we can use to decrease the bag of the center in such a way that this particular image is no longer in the torso of the center. Note that when we exclude some vertices from the bag, then new cliques can appear in the torso. The main technical challenge is to ensure that no new clique minor images are created when decreasing the size of the bag.

The third and final decomposition lemma takes a clique minor image I attached to an unbreakable set S and finds either a star decomposition where the torso of the center has "almost bounded degree" (that is, bounded degree with the exception of a bounded number of vertices) or a subdivision of a clique.

LEMMA 4.10 (BOUNDED-DEGREE STAR DECOMPOSITION). *For every $k \in \mathbb{N}$, there exist constants $c^*(k)$, $d^*(k)$, $m^*(k)$, $\ell^*(k)$ such that the following holds. There is an $f(k)|V(G)|^{O(1)}$ time algorithm that given a graph G, integer k, an m-unbreakable set X of size $3m-2$ (for $m := m^*(k)$) and an image I of K_ℓ that is m-attached to X (for $\ell := \ell^*(k)$),*

(1) either finds a subdivision of K_k in G,
(2) or computes a star decomposition $\Sigma_X = (T_X, \sigma_X, \alpha_X)$ of $G \cup K[X]$ having adhesion $< |X|$ such that $X \subseteq \beta(s)$ and at most $c^*(k)$ vertices of $\tau(s)$ have degree greater than $d^*(k)$ in $\tau(s)$, where s is the center of T_X.

Furthermore, suppose that the algorithm computes Σ_X on input (G, X) and let (G', X') be a pair such that there is an isomorphism f from G to G' with $f(X) = X'$. Then the algorithm computes a star decomposition $\Sigma'_{X'}$ on input (G', X') and there exists an isomorphism g from T_X to $T_{X'}$ such that for all $t \in V(T_X)$ we have $\sigma_{X'}(g(t)) = f(\sigma_X(t))$ and $\alpha_{X'}(g(t)) = f(\alpha_X(t))$.

The proof of Lemma 4.10 is deferred to Section 6.2. The main idea is that we are trying to remove every high-degree vertex from the bag of the center using appropriate separations. If there are at least k high-degree vertices that cannot be removed this way, then these vertices are close to the clique minor image I, and we can use this fact to construct a subdivision of a clique.

With the three local decomposition algorithms of Lemmas 4.7–4.10 at hand, we are ready to prove Local Structure Theorem 4.2:

PROOF PROOF OF LOCAL STRUCTURE THEOREM 4.2. Let $c(k) = c^*(k)$, $d(k) = d^*(k)$, $\ell = \ell(k) = \ell^*(k)$, $m = m(k) = m^*(k)$ using the functions c^*, d^*, ℓ^*, m^* in Lemma 4.10. Let $e(k) = e^*(\ell, m)$ for the function e^* in Lemma 4.9. Let $b(k) = b^*(m)$ for the function $b^*(k)$ in Lemma 4.7. Let $a(k) = 3m-3$. Note that $b^*(m) \geq 3m-3$ in Lemma 4.7: otherwise, neither (1) nor (2) would be possible if $X = V(G)$ and $|X| = 3m-3$. Thus we can assume $b(k) \geq a(k)$.

If $S = V(G)$, then we can return a star decomposition consisting of a single center node s with $\alpha(s) = V(G)$ and $\sigma(s) = \varnothing$ (here we use that $b(k) \geq a(k) \geq |S|$). Let $X := S \cup \{v\}$ for an arbitrary vertex $v \notin S$. Let us call the algorithm of Lemma 4.7 on G, X, and m. If it returns a star decomposition $\Sigma_X = (T_X, \alpha_X, \sigma_X)$, then we return it and we are done. Note that in this case $v \in X \subseteq \beta_X(s)$ for the root s of T_X, thus $v \notin \alpha_X(t)$ for any tip t of T_X, which means that the requirement $\alpha_X(t) \subset \alpha_X(s)$ indeed holds. Otherwise, let X' be the m-unbreakable superset of X returned by the algorithm. Let us call the algorithm of Lemma 4.9 with G, ℓ, m, and X'. Again, if it returns a star decomposition, we are done. Otherwise, it returns a K_ℓ-minor image I that is m-attached to I. Let us call the algorithm of Lemma 4.10 with G, k, X', and I. It returns either a K_k-subdivision or a star decomposition; we are done in both cases. □

5. TANGLES

In the proofs of the local decomposition lemmas (Section 6), we need to deal with separations that separate some set from (the larger part of) an unbreakable set. Robertson and Seymour [23] defined the abstract notion of *tangles,* which is a convenient tool for describing such separations. While in principle our results could be described without introducing tangles (in particular, we are not using any previous results about tangles), we feel that they provide a convenient notation for our purposes, and they make our results slightly more general.

Let $m \in \mathbb{N} \setminus \{0\}$. A *tangle of order m* in a graph G is a set $\mathfrak{T}$ of separations of G of order $< m$ such that the following axioms are satisfied:

(TA.1) For every separation (A, B) of G of order $< m$, either $(A, B) \in \mathfrak{T}$ or $(B, A) \in \mathfrak{T}$.

(TA.2) For all $(A_1, B_1), (A_2, B_2), (A_3, B_3) \in \mathfrak{T}$ it holds that $A_1 \cup A_2 \cup A_3 \neq G$.

(TA.3) For all $(A, B) \in \mathfrak{T}$ it holds that $V(A) \neq V(G)$.

Intuitively, one can think of each separation (A, B) in the tangle $\mathfrak{T}$ as having a "small side" A and "big side" B. Axiom (TA.2) states that the "small side" is so small that not even three of them can cover the whole graph.

In this paper, we only consider tangles of a special form. These tangles are defined by unbreakable sets (in the sense of Definition 4.5).

LEMMA 5.1. *Let X be an m-unbreakable set of size at least $(3m-2)$ in graph G. Let $\mathfrak{T}$ contain every separation of order $< m$ such that $|(X \cap V(B)) \cup V(A \cap B)| \geq |X|$. Then $\mathfrak{T}$ is a tangle of order m in G (and we call it the tangle of order m defined by the set X). Furthermore, for every separation $(A, B) \in \mathfrak{T}$ it holds that $|V(A) \cap X| \leq |V(A \cap B)| < m$.*

The size of a tangle (even of small order) can be exponential in the size of the graph. Observe that specifying the vertex set $V(A) \cap V(B)$ is not sufficient for describing the separation (A, B). For example, a star with n leaves have at least 2^n separations of order 1. Therefore, when stating algorithmic results that take a graph and a tangle as input, we have to state how the tangle is represented. To obtain maximum generality of the results, we assume that the tangle is given by an oracle. We define two type of oracles. The first type simply answers if a separation (A, B) is a member of the tangle. However, in applications we often need to find a separation of small order in the tangle that separates two given sets S and T. The min-cut oracle answers queries of this type. Note that there are more than one natural way of defining such oracles, in particular, we might want to allow or forbid the separator $V(A) \cap V(B)$ to intersect S and/or T. We define the min-cut oracle in a way that includes all these possibilities: the query contains a set F of forbidden vertices and we require the separator to be disjoint from F.

DEFINITION 5.2. *Let $\mathfrak{T}$ be a tangle of order k in a graph G.*

(1) An oracle *for $\mathfrak{T}$ answers in constant time whether a given separation (A,B) is in $\mathfrak{T}$.*
(2) Given sets $S,T,F \subseteq V(G)$ and an integer $\lambda < k$, a min cut oracle *for $\mathfrak{T}$ returns in constant time either a separation $(A,B) \in \mathfrak{T}$ of order at most λ such that $S \subseteq V(A)$, $T \subseteq V(B)$, and $V(A) \cap V(B) \cap F = \emptyset$, or "no" if no such separation exists.*

For tangles defined by unbreakable sets it is easy to implement both type of oracles:

LEMMA 5.3. *Let X be an m-unbreakable set of size at least $3m-2$ in a graph G and let $\mathfrak{T}$ be the tangle of order m defined by X.*

(1) The oracle for $\mathfrak{T}$ can be implemented in polynomial time.
(2) The min cut oracle for $\mathfrak{T}$ can be implemented in time $2^{|X|} \cdot |V(G)|^{O(1)}$.

5.1 Boundaries and separations

In this section, we summarize some useful properties of boundaries of sets and their relations to tangles. These facts will be used extensively in Section 6.

Recall that $\partial^G(X) = |N^G(X)|$. The following lemma states that the function ∂ satisfies the submodular inequality and a variant of the posimodular inequality:

LEMMA 5.4. *Let G be a graph and $X,Y \subseteq V(G)$.*

(1) $\partial(X) + \partial(Y) \geq \partial(X \cap Y) + \partial(X \cup Y)$.
(2) $\partial(X) + \partial(Y) \geq \partial(X \setminus N^G[Y]) + \partial(Y \setminus N^G[X])$.

We often work with separations that separate a subset of vertices from the rest of the graph:

DEFINITION 5.5. *Let G be a graph and $X \subseteq V(G)$. Then we define the separation $S_G(X) = (A,B)$ by $A = G[N^G[X]]$, $V(B) = V(G) \setminus X$, $E(B) = E(G) \setminus E(A)$.*

Note that the order of $S_G(X)$ is exactly $\partial^G(X)$.

The following observation, together with Lemma 5.4, will allow us to use uncrossing arguments in Section 6:

LEMMA 5.6. *Let $\mathfrak{T}$ be a tangle of order m in graph G and let $X,Y \subseteq V(G)$ be sets such that $S_G(X), S_G(Y) \in \mathfrak{T}$.*

(1) For every $X' \subseteq X$, if $S_G(X')$ is of order $< m$, then $S_G(X') \in \mathfrak{T}$.
(2) If $S_G(X \cap Y)$ is of order $< m$, then $S_G(X \cap Y) \in \mathfrak{T}$.
(3) If $S_G(X \cup Y)$ is of order $< m$, then $S_G(X \cup Y) \in \mathfrak{T}$.

We say that a separation *removes* a set $X \subseteq V(G)$ if $X \subseteq V(A) \setminus V(B)$. Note that $S_G(W)$ removes X if and only if $X \subseteq W$. It follows from Lemmas 5.4 and 5.6 that for every set X, there is a unique "closest minimum cut" of the tangle that removes X:

LEMMA 5.7. *Let $\mathfrak{T}$ be a tangle of order m in a graph G. Suppose that $X \subseteq V(G)$ is a set such that there is a $W \subseteq V(G)$ with $X \subseteq W$ and $S_G(W) \in \mathfrak{T}$. Then there is a unique $W(X) \subseteq V(G)$ such that*

(1) $X \subseteq W(X)$,
(2) $S_G(W(X)) \in \mathfrak{T}$,
(3) the order of $S_G(W(X))$ is minimum possible, and
(4) among such sets, $|W(X)|$ is minimum possible.

Furthermore, given a min cut oracle for $\mathfrak{T}$, this unique minimal set can be found in polynomial time.

PROOF. Let $m_0 < m$ be the minimum possible order of a separation $S_G(W) \in \mathfrak{T}$ over all W containing X. To prove the uniqueness of $W(X)$, we show a stronger statement: there is such a $W(X)$ with the property that $W(X) \subseteq W$ for every $W \supseteq X$ with $S_G(W) \in \mathfrak{T}$ and $\partial(W) = m_0$. To prove this statement, suppose that $W_1, W_2 \supseteq X$ are sets such that $S_G(W_1), S_G(W_2) \in \mathfrak{T}$ both have order m_0. By Lemma 5.4(1),

$$2m_0 = \partial(W_1) + \partial(W_2) \geq \partial(W_1 \cap W_2) + \partial(W_1 \cup W_2).$$

Observe that $W_1 \cap W_2$ and $W_1 \cup W_2$ both contain X. If $\partial(W_1 \cup W_2) < m_0$, then $S_G(W_1 \cup W_2) \in \mathfrak{T}$ by Lemma 5.6(3), contradicting the minimality of the order of $S_G(W_1)$ and $S_G(W_2)$. If $\partial(W_1 \cup W_2) \geq m_0$, then $\partial(W_1 \cap W_2) \leq m_0$. By Lemma 5.6(2), $S_G(W_1 \cap W_2) \in \mathfrak{T}$, and its order is not larger than the order of $S_G(W_1)$ and $S_G(W_2)$. Thus the intersection of the two sets is also a set satisfying the requirements. It follows that the common intersection of every $W_i \supseteq X$ such that $\partial(W_i) = m_0$ and $S_G(W_i) \in \mathfrak{T}$ is the required minimal set $W(I)$.

To find this unique set $W(X)$, we let $S := X$, initially define $T = \emptyset$, and use the min cut oracle to check if there is a separation (A,B) of order at most λ with $X \subseteq V(A)$, $T \subseteq V(B)$, and $V(A) \cap V(B)$ disjoint from $F := X$. Let us fix the smallest λ for which the answer is yes: then the min cut oracle returns a separation $(A,B) \in \mathfrak{T}$, such that $W := V(A) \setminus V(B)$ satisfies the first three properties above. To ensure that the last property holds as well, we pick a vertex $v \in W$, and call the min cut oracle to check if there is a separation $(A',B') \in \mathfrak{T}$ of order λ such that $X \subseteq V(A')$, $T \cup \{v\} \subseteq V(B')$, and $V(A') \cap V(B')$ disjoint from X. If there is such a separation, then we include v in T, and repeat this process with the new separation (A',B'). As the size of T strictly increases, eventually we arrive at a set W such that including any vertex $v \in W$ into T increases the minimum cut size to above λ. We have seen that this set W contains the unique minimal set $W(X)$ defined above. Furthermore, $W = W(X)$ has to hold: otherwise, including a vertex $v \in W \setminus W(X)$ into T would not increase the minimum cut size. □

The following observation is immediate:

PROPOSITION 5.8. *If $G[X]$ is connected, then $G[W(X)]$ is connected.*

6. PROOFS OF THE LOCAL DECOMPOSITION LEMMAS

This section completes the proof of Global Structure Theorem 4.1 by proving Lemmas 4.9 and 4.10 (Sections 6.1 and 6.2). Note that the proofs in this section contain somewhat more work than what is strictly necessary for the proof of the Global Structure Theorem 4.1: the proof of the invariance conditions in Lemmas 4.9 and 4.10 require extra arguments. These invariance conditions are not needed for the Global Structure Theorem, but they will be crucial for the invariance of the treelike decompositions in Section 8 and therefore for the results of Section 9 on isomorphism and canonization.

We prove variants of Lemmas 4.9 and 4.10 stated in terms of tangles instead of unbreakable sets (Lemmas 6.9 and 6.11, respectively); the proofs of Lemmas 4.9 and 4.10 follows easily from these variants. The statements involving tangles need the following definitions:

DEFINITION 6.1. *Let $\mathfrak{T}, \mathfrak{T}'$ be tangles in graphs G, G', respectively. An* isomorphism *from $(G, \mathfrak{T})$ to $(G', \mathfrak{T}')$ is an isomorphism f from G to G' such that for all $(A, B) \in \mathfrak{T}$ we have $(f(A), f(B)) \in \mathfrak{T}'$.*

DEFINITION 6.2. *Let $\Sigma = (T, \beta)$ be a star decomposition of graph G and let $\mathfrak{T}$ be a tangle of G. We say that Σ* respects *$\mathfrak{T}$ if for every tip t of T the separation (A, B) with $A = G[\gamma(t)]$ and $V(B) = V(G) \setminus \alpha(t)$ is in $\mathfrak{T}$. In particular, this implies $S_G(\alpha(t)) \in \mathfrak{T}$ and $|\sigma(t)|$ is less than the order of $\mathfrak{T}$.*

A key tool in our proofs is the following lemma, which follows from [24, (5.3)]:

LEMMA 6.3 ([24]). *For every $r \in \mathbb{N}$, there is a constant $t(r) = O(r^2)$ such that the following holds. Let G be a graph and $R \subseteq V(G)$ with $|R| = r$. Let $t \geq t(r)$ and let $(B_i)_{i \in [t]}$ be an image of a K_t-minor in G. Suppose that there is no separation (G_1, G_2) of G of order $< |R|$ with $R \subseteq V(G_1)$ and $B_b \cap V(G_1) = \emptyset$ for some $b \in [t]$. Then there is a $K_{|R|}$-minor image in G such that every branch set contains exactly one vertex of R and such an image can be found in polynomial time.*

6.1 Star decomposition with clique-minor free center

We prove Lemma 4.9 in this section. First we prove a variant of the lemma stated in terms of tangles (Lemma 6.9) and then deduce Lemma 4.9 it at the end of the section.

Recall that a separation (A, B) removes a set X if $X \subseteq V(A) \setminus V(B)$. We say that a separation (A, B) *removes* the H-minor image $I = (I_w)_{w \in V(H)}$ if it removes one of the branch sets, that is, $I_w \subseteq V(A) \setminus V(B)$ for some $w \in V(H)$. A tangle $\mathfrak{T}$ in G *removes an H-minor image I* if I is removed by some $(A, B) \in \mathfrak{T}$ with order $< |H|$. The following lemma is analogous to Lemma 5.7: for every clique minor, there is a unique "closest minimum separation" that removes it.

LEMMA 6.4. *Let $\mathfrak{T}$ be a tangle of order m in a graph G and let $e > 2m$. For every image I of K_e in G removed by $\mathfrak{T}$, there is a unique $W(I) \subseteq V(G)$ such that*

(1) $S_G(W(I))$ removes I,
(2) $S_G(W(I)) \in \mathfrak{T}$,
(3) the order of $S_G(W(I))$ is minimum possible, and
(4) among such sets, $|W(I)|$ is minimum possible.

Furthermore, $G[W(I)]$ is connected and there is a polynomial-time algorithm that, given G, m, I, and a min cut oracle for $\mathfrak{T}$, either finds $W(I)$ or concludes that $\mathfrak{T}$ does not remove I.

PROOF. As $\mathfrak{T}$ removes I, there has to be at least one separation $(A, B) \in \mathfrak{T}$ that removes I. Thus the set $W = V(A) \setminus V(B)$ is one such set. To prove the uniqueness, suppose that there are two distinct minimal sets X and Y. By Lemma 5.4(1), either $\partial(X \cap Y) \leq \partial(X)$ or $\partial(X \cup Y) < \partial(Y)$.

Suppose first that $\partial(X \cap Y) \leq \partial(X) < m$. By Lemma 5.6(2), $S_G(X \cap Y) \in \mathfrak{T}$. We claim that $S_G(X \cap Y)$ removes I. As both $S_G(X)$ and $S_G(Y)$ remove I, there are vertices $x, y \in V(K_e)$ such that $V(I_x) \subseteq X$ and $V(I_y) \subseteq Y$. Since $\partial(X), \partial(Y) < m$ and $e > 2m$, there is a vertex $z \in K_e$ such that $V(I_z)$ is disjoint from $N^G(X) \cup N^G(Y)$. As K_e is a clique, a vertex of $V(I_z)$ has to be adjacent to $V(I_x) \subseteq X$, which is only possible if this vertex is also in X (since it cannot be in $N^G(X)$). It follows that $V(I_z)$ is fully contained in X. A symmetrical argument shows that $V(I_z) \subseteq Y$. Thus $V(I_z) \subseteq X \cap Y$, i.e., $S_G(X \cap Y)$ removes I. Therefore, $X \cap Y \subset X$ and $\partial(X \cap Y) \leq \partial(X)$ contradicts the minimality of X.

Suppose now that $\partial(X \cup Y) < \partial(Y) < m$. By Lemma 5.6(3), $S_G(X \cup Y) \in \mathfrak{T}$. Clearly, $S_G(X \cup Y)$ removes I (as any branch set contained in X or Y is also contained in $X \cup Y$). Therefore, $X \cup Y$ contradicts the minimality of Y.

To check if an image I is removed by $\mathfrak{T}$, we use the algorithm of Lemma 5.7 to compute the set $W(I_v)$ for every $v \in V(K_e)$ (if such a set exists). If $\mathfrak{T}$ removes I, then at least one of these sets should exist. Furthermore, if $\mathfrak{T}$ removes I, then it should be clear that $W(I)$ is equal to one of these sets $W(I_v)$: if $W(I)$ contains I_v, then it cannot be different from $W(I_v)$ (as it would contradict the minimality and uniqueness of either $W(I)$ or $W(I_v)$). As $W(I_v)$ is connected by Prop. 5.8, it follows that $W(I)$ is connected as well. □

A simple uncrossing argument shows that the minimum separations defined in Lemma 6.4 cannot properly intersect each other:

LEMMA 6.5. *Let $\mathfrak{T}$ be a tangle of order m in a graph G and let $e > 2m$. Let I^x and I^y be two K_e-minor images removed by $\mathfrak{T}$. Then either*

(1) $W(I^x) \subseteq W(I^y)$,
(2) $W(I^x) \supseteq W(I^y)$, or
(3) $W(I^x)$ and $W(I^y)$ are disjoint and do not touch.

PROOF. Let $X := W(I^x)$ and $Y := W(I^y)$ and suppose that none of the three possibilities hold. Assume first that I^x has a branch set fully contained in $X \cap Y \subset X$. If $\partial(X \cap Y) \leq \partial(X) < m$, then $S_G(X \cap Y) \in \mathfrak{T}$ by Lemma 5.6(2) and $S_G(X \cap Y)$ removes I^x, contradicting the minimality of $W(I^x)$. Thus we can assume that $\partial(X \cap Y) > \partial(X)$. By Lemma 5.4, it follows that $\partial(X \cup Y) < \partial(Y) < m$. Therefore, $S_G(X \cup Y)$ is in $\mathfrak{T}$ by Lemma 5.6(3) and it clearly removes I^y (since $S_G(Y)$ already does), contradicting the minimality of $Y = W(I^y)$.

We have proved that I^x has no branch set fully contained in $X \cap Y$, and a symmetrical argument shows that I^y has no such branch set either. By Lemma 5.4(2), either $\partial(X) \geq \partial(X \setminus N[Y])$ or $\partial(Y) \geq \partial(Y \setminus N[X])$. Assume without loss of generality the first case. Consider a branch set I^x_1 of I^x fully contained in X (such a set exists, as $S_G(X)$ removes I^x) and a branch set I^x_2 disjoint from $N^G(X) \cup N^G(Y)$ (since $e > 2m$, there has to be such a set). The branch set I^x_2 has a vertex adjacent to $I^x_1 \subseteq X$. Since I^x_2 is disjoint from $N^G(X)$, this is only possible if I^x_2 is fully contained in X. Moreover, we assumed that I^x_2 is disjoint from $N^G(Y)$ and it is not fully contained in $X \cap Y$, thus I^x_2 is fully contained in $X \setminus N^G[Y]$, that is, the separation $S_G(X \setminus N^G[Y])$ removes I^x. Note that $X \setminus N^G[Y]$ is a proper subset of X, otherwise X and Y are disjoint and do not touch. Lemma 5.6(1) implies that $S_G(X \setminus N[Y]) \in \mathfrak{T}$, and therefore $X \setminus N[Y] \subset X$ violates the minimality of $X = W(I^x)$. □

Another useful property of the definition of minimum separation in Lemma 6.4 is that if $S_G(W(I)) = (A, B)$, then the clique minor I allows us to connect vertices of $V(A) \cap V(B)$

with each other using paths in A in an arbitrary way. We use the following definition to state this property:

DEFINITION 6.6. *We say that a separation (A,B) of order m is* generic *if there is a K_m-minor image in A such that each branch set contains exactly one vertex of $V(A)\cap V(B)$. Such an image is called a* witness.

LEMMA 6.7. *Let $\mathfrak{T}$ be a tangle of order m in a graph G and let $e > t(m)+m$ for the function t of Lemma 6.3. For every image I of K_e in G removed by $\mathfrak{T}$, the separation $S_G(W(I))$ is generic. Furthermore, given I and a min cut oracle for $\mathfrak{T}$, a witness can be found in polynomial time.*

PROOF. Let $S_G(W(I)) = (A,B)$ and $R = V(A)\cap V(B)$. By definition, (A,B) removes I, thus at least one branch set of I is contained in $V(A)\setminus V(B)$ and at most $|R| < m$ branch sets intersect R. Thus at least $t(m)$ branch sets are fully contained in $V(A)\setminus V(B)$. Therefore, A contains a $K_{t(m)}$-minor image I'. We verify that the conditions of Lemma 6.3 hold for graph A and set R. Suppose that there is a separation (G_1,G_2) of order $< |R|$ with $R\subseteq G_1$ and $I'_w\subseteq V(G_2)\setminus V(G_1)$ for some branch set I'_w of I' (which is also a branch set of I). Let $X' = V(G_2)\setminus V(G_1)\subset V(A)\setminus V(B) = W(I)$. It follows that $S_G(X')$ has order $< |R|$ (which is the order of (A,B)) and is in $\mathfrak{T}$ by Lemma 5.6(1). However, $S_G(X')$ also removes I, contradicting the minimality of $W(I)$. We can conclude that A and R satisfy the conditions of Lemma 6.3, and the existence of the required $K_{|R|}$-minor image follows. □

It follows from Lemma 6.7 that if $W(I) = (A,B)$, then removing $V(A)\setminus V(B)$ and replacing $V(A)\cap V(B)$ with the clique $K[V(A)\cap V(B)]$ does not create any new clique minor images in B (because the edges in the clique $K[V(A)\cap V(B)]$ can be simulated by connections in A in the original graph). Repeated application of this observation shows that after removing all the clique minor images, we get a bag whose torso does not contain clique minors of the given size.

LEMMA 6.8. *Let $\mathfrak{T}$ be a tangle of order m in a graph G and let $e > t(m)+m$ for the function t of Lemma 6.3. Let $I^1,\ldots,I^p$ be K_e-minor images removed by $\mathfrak{T}$. Let $W = \bigcup_{i=1}^p W(I^i)$ and let $G' = \operatorname{torso}(G,V(G)\setminus W)$. The graph G' has a K_e-minor I' if and only if G has a K_e-minor image I not removed by any $S_G(W(I^i))$. Furthermore, given a min cut oracle for $\mathfrak{T}$ and such a K_e-minor image I, one can compute a K_e-minor image I' in G' in polynomial time and vice versa.*

PROOF. We can assume that the sets $W(I^1),\ldots,W(I^m)$ are pairwise incomparable (because if $W(I^i)\subseteq W(I^j)$, then omitting I^i from this collection does not change W), thus by Lemma 6.5, we can assume that these sets are pairwise disjoint and do not touch. This means that $R_i = N^G(W(I^i))$ is a subset of $V(G)\setminus W$ and induces a clique in G'. By Lemma 6.4, each $G[W(I^i)]$ is connected. Thus $G' = \operatorname{torso}(G,V(G)\setminus W)$ is exactly the union of $G\setminus W$ with a clique on each R_i.

Let I' be the image of a K_e-minor in G'. Note that this is not necessarily a K_e-minor image in $G\setminus W$ as G' has edges that $G\setminus W$ do not have. However, we can use the subgraph inside $G[W(I^i)]$ to simulate these edges. By Lemma 6.7, every $S_G(W(I^i))$ is generic and we can obtain the corresponding clique minor images. This means that for each R_i, there is a set of r pairwise disjoint and touching connected subgraphs in $G[N^G[W(I^i)]]$. Using these connected sets, we can extend each I'_w of G' into a connected set I_w of G and obtain a K_e-minor image I in G.

For the reverse direction, let I be a K_e-minor image in G not removed by any $S_G(W(I^i))$. Let I' be defined by $I'_w = G'[V(I_w)\setminus W]$ for every $w\in V(K_e)$. Note that $V(I'_w)\neq\emptyset$: this would be only possible if $V(I_w)\subseteq W(I^i)$ for some $1\le i\le p$, which would imply that $S_G(W(I^i))$ removes I. We claim that I' is a K_e-minor image. The connectedness of I'_w is easy to see: any path with internal vertices in $W(I^i)$ can be replaced by an edge in R_i (as R_i induces a clique in G'). To see that I'_w and I'_u touch for every $w,u\in V(K_e)$, consider an edge e between I_w and I_u in G. If both endpoints of e are in $W(I^i)\cup R_i$, then I'_w and I'_u both intersect R_i, hence they touch. Otherwise, e is an edge of $G\setminus W$, implying that it is also an edge of G'. □

Now we state and prove a version of Lemma 4.9 in terms of tangles:

LEMMA 6.9. *For every $\ell,m\in\mathbb{N}$, there is a constant $e'(\ell,m)$ such that the following holds. There is an $f(\ell,m)\cdot|V(G)|^{O(1)}$ time algorithm that, given a graph G, ℓ, m, a min cut oracle for a tangle $\mathfrak{T}$ of order m, either*

(1) finds a K_ℓ-minor image I not removed by $\mathfrak{T}$, or
(2) computes a $\mathfrak{T}$-respecting star decomposition $\Sigma_{\mathfrak{T}} = (T_{\mathfrak{T}},\alpha_{\mathfrak{T}},\sigma_{\mathfrak{T}})$ with center s such that $\tau_{\mathfrak{T}}(s)$ does not contain a $K_{e'(\ell,m)}$-minor.

Furthermore, if the algorithm returns $\Sigma_{\mathfrak{T}}$ for $(G,\mathfrak{T})$ and $\mathfrak{T}'$ is another tangle of order m in a graph G', and f is an isomorphism from $(G,\mathfrak{T})$ to $(G',\mathfrak{T}')$, then the algorithm returns a star decomposition $\Sigma_{\mathfrak{T}'}$ for $(G',\mathfrak{T}')$ such that there is an isomorphism g from $T_{\mathfrak{T}}$ to $T_{\mathfrak{T}'}$ such that for all $t\in V(T_{\mathfrak{T}})$ we have $\sigma_{\mathfrak{T}'}(g(t)) = f(\sigma_{\mathfrak{T}}(t))$ and $\alpha_{\mathfrak{T}'}(g(t)) = f(\alpha_{\mathfrak{T}}(t))$.

PROOF. Let $e = e'(\ell,m) = \max(\ell,t(m)+m+1)$ for the function t in Lemma 6.3. We show first that if $\mathfrak{T}$ removes every K_ℓ-minor image (and therefore every K_e-minor image as $e\ge\ell$), then there exists a star decomposition satisfying the requirements. Suppose that $\mathfrak{T}$ removes every K_ℓ-minor image, implying that $W(I)$ is defined for every K_e-minor image I. Let $I^1,\ldots,I^p$ be the list of all K_e-minor images for which $W(I^i)$ is inclusionwise maximal. By Lemma 6.5, $W(I^i)$ and $W(I^j)$ are disjoint and do not touch for $i\neq j$. Let $W = \bigcup_{i=1}^p W(I^i)$. We construct a star decomposition $\Sigma_{\mathfrak{T}} = (T_{\mathfrak{T}},\sigma_{\mathfrak{T}},\alpha_{\mathfrak{T}})$ with center s and p tips t_i $(1\le i\le p)$. We set $\alpha_{\mathfrak{T}}(s) = V(G)$, $\sigma_{\mathfrak{T}}(s) = \emptyset$, $\alpha_{\mathfrak{T}}(t_i) = W(I^i)$, and $\sigma_{\mathfrak{T}}(t_i) = N^G(W(I^i))$.

It easy easy to verify that Δ is a tree decomposition:

Claim 1. Δ satisfies properties (TD.1)–(TD.5).

The definition of $W(I^i)$ implies that $S_G(W(I^i))\in\mathfrak{T}$ for every $1\le i\le p$. Therefore,

Claim 2. Δ respects $\mathfrak{T}$.

Claim 3. $G' = \tau(s) = \operatorname{torso}(G,V(G)\setminus W)$ does not contain a K_e.

Proof. If G' contains a K_e-minor, then Lemma 6.8 implies that there is a K_e-minor image I in G not removed by any of the separations $S_G(W(I^i))$. However, this contradicts the assumption that $I^i,\ldots,I^p$ is the list of all images for which $W(I^i)$ is inclusionwise maximal.

Algorithmically, we can find the set W defined above as follows. We construct collections $\mathcal{I}^{(0)} \subset \mathcal{I}^{(1)} \subset \dots$ of K_e-minor images, each of which is removed by $\mathfrak{T}$. We start with $\mathcal{I}^{(0)} = \emptyset$. Given $\mathcal{I}^{(j)}$, we construct $\mathcal{I}^{(j+1)}$ as follows. Let $W^{(j)} = \bigcup_{I \in \mathcal{I}^{(j)}} W(I)$ and $G^{(j)} = \mathrm{torso}(G, V(G) \setminus W^{(j)})$. We test if $G^{(j)}$ has a K_e-minor (using the algorithm of Theorem 2.1). By Lemma 6.8, if there is a K_e-minor model I' in $G^{(j)}$, then there is a corresponding K_e-minor model $I^{(j)}$ of G which is not removed by $S_G(W(I))$ for any $I \in \mathcal{I}^{(j)}$. Let us use the algorithm of Lemma 6.4 to compute the set $W(I^{(j)})$. If the algorithm returns that $W(I^{(j)})$ is not defined, that is, $I^{(j)}$ is not removed by $\mathfrak{T}$, then we can stop and return $I^{(j)}$ (or more precisely, as $e \geq \ell$, a restriction of $I^{(j)}$ to a K_ℓ-minor) and we are done. Otherwise, let us obtain $\mathcal{I}^{(j+1)}$ from $\mathcal{I}^{(j)}$ by inserting $I^{(j)}$. Let us observe that $W(I^{(j)}) \notin W^{(j)}$: by Lemma 6.5, $W(I^{(j)}) \subseteq W^{(j)}$ is only possible if $W(I^{(j)}) \subseteq W(I)$ for some $I \in \mathcal{I}^{(j)}$, but this means that $S_G(W(I))$ already removes $I^{(j)}$, a contradiction. It follows that $W^{(j)} \subset W^{(j+1)}$. After including $I^{(j)}$ into $\mathcal{I}^{(j+1)}$, we repeat this procedure until we arrive to a j such that $G^{(j)}$ has no K_e-minor.

As the size of $W^{(j)}$ strictly increases in each step, the process described above stops in at most $|V(G)|$ steps with a $G^{(j)}$ that does not contain a K_e-minor.

Claim 4. $W^{(j)} = W$.

Proof. Suppose that $W^{(j)} \neq W$, i.e., there is an image I^* such that $W(I^*) \notin W^{(j)}$. Since $G^{(j)}$ has no K_e-minor, by Lemma 6.8, there is an $I \in \mathcal{I}^{(j)}$ such that $S_G(W(I))$ removes I^*. As $S_G(W(I))$ and $S_G(W(I^*))$ both remove I^*, the sets $W(I)$ and $W(I^*)$ both contain a branch set of I^*, hence it is not possible that the two sets are disjoint and do not touch. Therefore, by Lemma 6.5, one of the two sets is contained in the other. From $W(I^*) \notin W^{(j)}$, we know that $W(I^*) \subseteq W(I)$ is not possible, hence we have $W(I) \subset W(I^*)$, implying that $S_G(W(I^*))$ removes I as well. Now $\partial(W(I^*)) < \partial(W(I))$ would contradict the minimality of $W(I)$ and $\partial(W(I^*)) \leq \partial(W(I))$ would contradict the minimality of $W(I^*)$ (as $|W(I)| < |W(I^*)|$). Thus we have proved that $W^{(j)}$ obtained by this procedure is indeed the set W defined at the beginning of the proof. ⌟

What remains to be proven is the invariance condition. Suppose that $\mathfrak{T}'$ is another tangle of order m in a graph G'. Let f be an isomorphism from $(G, \mathfrak{T})$ to $(G', \mathfrak{T}')$. Let $I = (I_v)_{v \in V(K_\ell)}$ be a K_e-minor image in G and let $I' = (f(I_v))_{v \in V(K_e)}$ be the corresponding K_e-minor image in G'. Let $W(I)$ and $W'(I')$ be the set given by Lemma 6.4 on I and I', respectively.

Claim 5. $W'(I') = f(W(I))$.

Proof. The definition of the set $W(I)$ depends only on the branch sets of I, the tangle $\mathfrak{T}$ and the graph-theoretical properties of G (size of the boundaries of certain sets etc.) and all these properties are preserved by f. ⌟

Therefore, if $\{W(I^1), \dots, W(I^p)\}$ is the collection of inclusionwise maximal sets appearing in the definition of W for $(G, \mathfrak{T})$, then exactly $\{f(W(I^1)), \dots, f(W(I^p))\}$ is the collection of sets appearing in the definition of W'. If follows that for every t_i, there is a $g(t_i)$ such that $\alpha_{\mathfrak{T}}(t_i) = W(I^i)$ and $\alpha_{\mathfrak{T}'}(g(t_i)) = f(W(I^i))$. Moreover, $\sigma_{\mathfrak{T}}(t_i) = N^G(W(I^i))$ and $\sigma_{\mathfrak{T}'}(g(t_i)) = N^{G'}(f(W(I^i))) = f(N^G(W(I^i)))$ follows, as required. Setting $g(s) = s'$ (where s' is the center of the decomposition of G') completes the definition of g. □

Finally, we can prove Lemma 4.9 by invoking Lemma 6.9 on the tangle defined by the unbreakable set X:

Proof (of Lemma 4.9). Let $e^*(\ell, m) = e'(\ell, m) + 3m - 2$ for the function x' in Lemma 6.9. Let $\mathfrak{T}$ be the tangle of order m defined by the m-unbreakable set X; Lemma 5.1 provides an implementation of the min-cut oracle for $\mathfrak{T}$. Let us call the algorithm of Lemma 6.9 with G, $\mathfrak{T}$, ℓ, and m. If it returns a K_ℓ-minor image I not removed by $\mathfrak{T}$, then this is equivalent to saying that I is m-attached to X. Thus we can return I and we are done. Otherwise, the algorithm of Lemma 6.9 returns a $\mathfrak{T}$-respecting star decomposition $\Sigma_{\mathfrak{T}} = (T_{\mathfrak{T}}, \alpha_{\mathfrak{T}}, \sigma_{\mathfrak{T}})$ of G. We construct a star decomposition $\Sigma_X = (T_X, \alpha_X, \sigma_X)$ as follows. First, let $T_X = T_{\mathfrak{T}}$ and for the center s of T_X, let $\alpha_X(s) = V(G)$ and $\sigma_X(s) = \emptyset$. For every tip t of T_X, we let $\alpha_X(t) = \alpha_{\mathfrak{T}}(t) \setminus X$ and $\sigma_X(t) = \sigma_{\mathfrak{T}}(t) \cup (X \cap \alpha_{\mathfrak{T}}(t))$. It is straightforward to verify that Σ_X is also a star decomposition of G, and in fact it is star decomposition even for the supergraph $G \cup K[X]$ (since $X \subseteq \beta_X(s)$). Note that $\tau_{\mathfrak{T}}(s) \setminus X = \tau_X(s) \setminus X$ (because the two bags differ only in the vertices of X and all the extra edges of $G \cup K[X]$ are incident to X). As $\tau_{\mathfrak{T}}(s)$ has no $K_{e'(\ell,m)}$-minor, this means that $\tau_X(s)$ cannot have a clique minor of order $e'(\ell, m) + |X| = e^*(\ell, m)$, as required. Furthermore, as $\Sigma_{\mathfrak{T}}$ is $\mathfrak{T}$-respecting, it follows that $|\sigma_{\mathfrak{T}}(t)| < m$ and $S_G(\alpha_{\mathfrak{T}}(t)) \in \mathfrak{T}$ for every tip t. By Lemma 5.1, this also means that $|\alpha_{\mathfrak{T}}(t) \cap X| \leq m - 1$ and therefore $|\sigma_X(t)| \leq m - 1 + m - 1 < |X|$. Thus the adhesion of Σ_X is less than $|X|$, as required.[1] The invariance condition follows easily from the invariance condition of Lemma 6.9: if f is an isomorphism from G to G' with $f(X) = X'$ and $\mathfrak{T}$ and $\mathfrak{T}'$ are the tangles defined by the unbreakable sets X and X', respectively, then f is an isomorphism from $(G, \mathfrak{T})$ to $(G', \mathfrak{T}')$. □

6.2 Star decomposition with a bounded-degree center

The proof of Lemma 4.10 has the same high-level strategy as the proof of Lemma 4.9 in Section 6.1: we identify those parts of the graph that we want to exclude from the bag of the center (this time, the high-degree vertices) and we use an uncrossing argument to show that all of them can be removed more or less independently from each other. The uncrossing argument is somewhat more involved due to the technicality that a high-degree vertex can be part of the separator removing some other high-degree vertex.

First we need the following lemma, which shows that all but at most k high-degree vertices can be removed by separations in the tangle, or we can find a K_k-subdivision.

[1] This is the point (and the analogous argument in the proof of Lemma 4.10) where it becomes motived why we used the tangle $\mathfrak{T}$ defined by the unbreakable set X. If we have no bound on $|\alpha_{\mathfrak{T}}(t) \cap X|$, then moving X to the center can increase the adhesion by up to $|X| = 3m - 2$, which means that the bound on the adhesion would be larger than $|X|$. Therefore, the repeated application of this lemma in the proof of Global Structure Theorem 4.1 would increase the adhesion in each step. In all the arguments in the section, we were careful enough to use only separations that are in $\mathfrak{T}$, and therefore we have the bound that the component of each child of t contains at most $m - 1$ vertices of X.

LEMMA 6.10. *For every $k \in \mathbb{N}$, there is a constant $\ell'(k)$ such that the following holds. For a graph G, integer $k \in \mathbb{N}$, tangle $\mathfrak{T}$ of order at least $k(k-1)$, and an image I of $K_{\ell'(k)}$ not removed by $\mathfrak{T}$, let Z contain a vertex $v \in V(G)$ if v has degree at least k and either $W(\{v\})$ is undefined or $\partial(W(\{v\})) \geq k(k-1)$. If $|Z| \geq k$, then given G, k, a min-cut oracle for $\mathfrak{T}$, and I, a subdivision of K_k in G can be found in polynomial time.*

PROOF. Let $\ell'(k) = t(k(k-1))$ for the function t appearing in Lemma 6.3. We show that if $|Z| \geq k$, then we can find a subdivision of K_k in G. Let Z_0 be a subset of Z of size exactly k. Let G' be the graph obtained from G by extending each vertex $z \in Z_0$ into a clique K_z of $k-1$ vertices: for every $z \in Z_0$, we introduce $k-2$ new vertices that are adjacent to each other, to vertex z, and to every neighbor of z. The clique K_z contains z and these $k-2$ new vertices. Let $R := \bigcup_{z \in Z_0} K_z$.

Let $I_1, \ldots, I_\ell$ be the branch sets in the K_ℓ minor image I. Let us show first that the conditions of Lemma 6.3 hold for R in G'. Suppose for contradiction that (A', B') is a separation of G' of order less than $|R| = k(k-1)$ with $R \subseteq V(A')$ and $I_b \subseteq V(B') \setminus V(A')$ for some $b \in [\ell]$. Let $Q' := V(A') \cap V(B')$ be the separator. Without loss of generality, we may assume that for all $z \in Z_0$, either $K_z \cap Q' = \varnothing$ or $K_z \subseteq Q'$. Let $A := A' \setminus (R \setminus Z_0)$ and $B := B' \setminus (R \setminus Z_0)$ (i.e., we remove from Q' the extra vertices that were introduced in the definition of G'). Then (A, B) is a separation of G; let $Q = V(A) \cap V(B)$ be the separator. Now it is clear that $|Q| \leq |Q'| < k(k-1)$. Furthermore, there has to be a vertex $z \in Z_0$ which is not in Q: otherwise, $Z_0 \subseteq Q$ implies that the size of Q' in G' is at least $k(k-1)$. Therefore, (A, B) is a separation of order $< k(k-1)$ with $z \in V(A) \setminus V(B)$. This separation is in $\mathfrak{T}$: otherwise, $(B, A) \in \mathfrak{T}$ by (TA.1) (here we use that the order of $\mathfrak{T}$ is at least $k(k-1)$) and $I_b \subseteq V(B)$ means that $\mathfrak{T}$ removes I, contradicting our assumption on I. It follows that $(A, B) \in \mathfrak{T}$ is a separation of order $< k(k-1)$ with $z \in V(A) \setminus V(B)$, contradicting $z \in Z$ and the definition of Z. Thus we can conclude that there is no such separation (G'_1, G'_2) of G', and the conditions of Lemma 6.3 hold for Z and G'.

Lemma 6.3 gives us a $K_{k(k-1)}$-minor image, that is, for every $q \in R$, a connected set I_q such that these sets are pairwise disjoint and touch. Consider a partition of R into $\binom{k}{2}$ classes, each of size 2, such that for every pair $z_1, z_2 \in Z_0$ of distinct vertices, there is a class of the partition containing a vertex of $\hat{z}_1 \in K_{z_1}$ and a vertex of $\hat{z}_2 \in K_{z_2}$. (As the size of each K_z is exactly $k-1$, such a partition is possible.) We define a path $P'_{\{z_1,z_2\}} \subseteq I_{\hat{z}_1} \cup I_{\hat{z}_2}$ connecting $\hat{z}_1$ and $\hat{z}_2$; let $\mathcal{P}'$ be the collection of these $\binom{k}{2}$ paths. For each such path $P'_{\{z_1,z_2\}} \in \mathcal{P}'$ of G', there is a corresponding path $P_{\{z_1,z_2\}}$ in G: whenever $P'_{\{z_1,z_2\}}$ contains a vertex of some K_z, then we replace it by z. Let $\mathcal{P}$ be the collection of these $\binom{k}{2}$ paths in G. As the paths $\mathcal{P}'$ are pairwise disjoint, the corresponding paths in $\mathcal{P}$ can intersect only in Z_0. Therefore, we have k vertices Z_0 and a collection of $\binom{k}{2}$ internally pairwise disjoint paths that connect every pair of vertices in Z_0. In other words, we have formed a K_k topological minor image in G, which we can return. □

The following lemma is a version of Lemma 4.10 stated in terms of tangles:

LEMMA 6.11. *For every integer $k \in \mathbb{N}$, there are constants $d'(k), m'(k), \ell'(k)$ such that the following holds. There is a polynomial-time algorithm that, given a graph G, an integer k, min cut oracle for a tangle $\mathfrak{T}$ of order $m'(k)$, and an image I of $K_{\ell'(k)}$ not removed by $\mathfrak{T}$, either*

(1) finds a subdivision of K_k in G, or
(2) computes a $\mathfrak{T}$-respecting star decomposition $\Sigma_{\mathfrak{T}} = (T_{\mathfrak{T}}, \sigma_{\mathfrak{T}}, \alpha_{\mathfrak{T}})$ of G with center s such that at most k vertices of $\tau_{\mathfrak{T}}(s)$ have degree more than $d'(k)$.

Furthermore, if the algorithm returns $\Sigma_{\mathfrak{T}}$ for $(G, \mathfrak{T})$ and $\mathfrak{T}'$ is another tangle of order m in a graph G', and f is an isomorphism from $(G, \mathfrak{T})$ to $(G', \mathfrak{T}')$, then the algorithm returns a star decomposition $\Sigma_{\mathfrak{T}'}$ for $(G', \mathfrak{T}')$ such that there is an isomorphism g from $T_{\mathfrak{T}}$ to $T_{\mathfrak{T}'}$ such that for all $t \in V(T_{\mathfrak{T}})$ we have $\sigma_{\mathfrak{T}'}(g(t)) = f(\sigma_{\mathfrak{T}}(t))$ and $\alpha_{\mathfrak{T}'}(g(t)) = f(\alpha_{\mathfrak{T}}(t))$.

PROOF. Let $\ell'(k)$ be as in Lemma 6.10. We will define later (in Claim 4) a constant a depending on k; let $m'(k) = \max\{k(k-1), a+1\}$. Let Z contain a vertex $v \in V(G)$ if v has degree at least k and $\partial(W(\{v\})) \geq k(k-1)$. If $|Z| \geq k$, then we can use the algorithm of Lemma 6.10 to return a subdivision of K_k in G, and we are done.

Otherwise, let $L \subseteq V(G)$ be the set of vertices not in Z having degree at least k. For every $v \in L$, let us use the algorithm of Lemma 5.7 to compute the unique minimal set $W_v = W(\{v\})$ (as $v \notin Z$, such a set exists). By Prop. 5.8, $G[W_v]$ is connected.

Let $\mathcal{W}$ contain the inclusionwise-maximal sets in $\{W_v \mid v \in L\}$; i.e., $W_v \in \mathcal{W}$ if and only if there is no $u \in L$ with $W_v \subset W_u$. Note that we define $\mathcal{W}$ such that it does not contain duplicate sets.

Claim 1. Every $b \in V(G)$ appears in $O(k^2)$ members of $\mathcal{W}$.

Proof. For every $W \in \mathcal{W}$, let us choose a representative $v \in L$ with $W = W_v$; let $M \subseteq L$ be the set of selected representatives. We define a directed graph $\overrightarrow{H}$ on M where $\overrightarrow{vu} \in E(\overrightarrow{H})$ if and only if $u \in N^G(W_v)$. Note that $|N^G(W_v)| < k(k-1)$ implies that the outdegree of v is at most $k(k-1)-1$. This further implies that the maximum clique size in the undirected graph H underlying $\overrightarrow{H}$ is at most $2k(k-1)-1$: the average degree of every subgraph of H is at most $2k(k-1)-2$.

We show that the representatives of the sets in $\mathcal{W}$ containing b form a clique in H, thus by the argument in the previous paragraph, there can be at most $2k(k-1)-1$ sets in $\mathcal{W}$ containing b. Consider two distinct vertices $u, v \in M$ with $b \in W_u$ and $b \in W_v$. We claim that u and v are adjacent in the undirected graph H. Otherwise, $u \notin N^G(W_v)$ and $v \notin N^G(W_u)$. We consider the following cases:

Case 1: $u \in W_u \cap W_v$. By Lemma 5.4(1), we have two possibilities:
(1) $\partial(W_u \cup W_v) < \partial(W_v)$. In this case $W_u \cup W_v$ contradicts the minimality of W_v (note that by Lemma 5.6(3), $S_G(W_u \cup W_v) \in \mathfrak{T}$).
(2) $\partial(W_u \cup W_v) \geq \partial(W_v)$ and $\partial(W_u \cap W_v) \leq \partial(W_u)$. In this case, $W_u \cap W_v$ contradicts the minimality of W_u (by Lemma 5.6(2), $S_G(W_u \cap W_v) \in \mathfrak{T}$).

Case 2: $v \in W_u \cap W_v$. Similar to case 1.

Case 3: $u \in W_u \setminus W_v$ and $v \in W_v \setminus W_u$. Let $W'_u := W_u \setminus N^G[W_v]$ and $W'_v := W_v \setminus N^G[W_u]$. Note that $b \in W_u \cap W_v$ implies that $W'_u \subset W_u$ and $W'_v \subset W_v$. Furthermore, the assumptions $u \in W_u \setminus W_v$ and $u \notin N^G(W_v)$ imply that $u \in W'_u$, and we have $v \in W'_v$ in a similar way. By Lemma 5.4(2), either $\partial(W'_u) \leq \partial(W_u)$ or $\partial(W'_v) \leq$

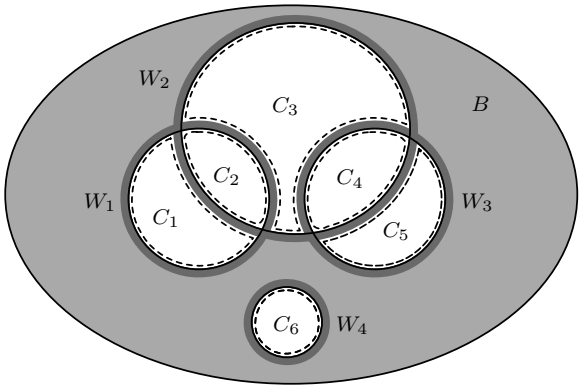

Figure 6.1: Definition of the set B in Lemma 6.11. The four solid circles represent the sets W_1, W_2, W_3, W_4 contained in $\mathcal{W}$. The dark gray area contains the boundaries of these sets. Set B (light and dark gray area) is defined to be the union of these boundaries and the area outside these sets. The six regions C_1, ..., C_6 with dashed outline are the components of $G \setminus B$.

$\partial(W_v)$. If, say, $\partial(W'_u) \leq \partial(W_u)$, then $S_G(W'_u) \in \mathfrak{T}$ follows by Lemma 5.6(1), contradicting the minimality of W_u.

Therefore, the vertices u of M for which $b \in W_u$ form a clique in H, thus there are less than $2k(k-1)$ such vertices. ⌟

We define

$$B := (V(G) \setminus \bigcup_{W \in \mathcal{W}} W) \cup \bigcup_{W \in \mathcal{W}} N^G(W)$$

(see Figure 6.1).

Claim 2. For every $W \in \mathcal{W}$, $|N^G[W] \cap B| = O(k^6)$.

Proof. Let us fix a $W \in \mathcal{W}$. We bound first the number of sets $Y \in \mathcal{W}$ such that $N^G(Y)$ intersects W. As $G[Y]$ is connected and Y is not contained in W (by the definition of $\mathcal{W}$), Y has to contain a vertex $b \in N^G(W)$. By Claim 1, there are at most $O(k^2)$ sets in $\mathcal{W}$ containing a particular $b \in N^G(W)$. Together with $|N^G(W)| < k(k-1)$, this gives a total bound of $O(k^4)$ on the number of sets $Y \in \mathcal{W}$ for which $N^G(Y)$ intersects W. As $|N^G(Y)| < k(k-1)$ for every $Y \in \mathcal{W}$, this means that W contains at most $O(k^6)$ vertices of B. Additionally, $N^G(W)$ can contain at most $|N^G(W)| < k(k-1)$ vertices of B, and the claim follows. ⌟

Let $C_1, \ldots, C_m$ be the connected components of $G \setminus B$. We construct a star decomposition $\Sigma_{\mathfrak{T}} = (T_{\mathfrak{T}}, \sigma_{\mathfrak{T}}, \alpha_{\mathfrak{T}})$ with center s and p tips t_i $(1 \leq i \leq p)$. We set $\alpha_{\mathfrak{T}}(s) = V(G)$, $\sigma_{\mathfrak{T}}(s) = \emptyset$, $\alpha_{\mathfrak{T}}(t_i) = C_i$, and $\sigma_{\mathfrak{T}}(t_i) = N^G(C_i)$.

It easy easy to verify that Δ is a tree decomposition:

Claim 3. Δ satisfies properties (TD.1)–(TD.5).

The following claim implies a bound on the adhesion of Δ:

Claim 4. There is a constant $a = O(k^6)$ such that $|\sigma_{\mathfrak{T}}(t_i)| \leq a$ for every $1 \leq i \leq p$.

Proof. The definition of B implies that for every $t \notin B$, there is a $W \in \mathcal{W}$ with $t \in W$. As $N^G(W) \subseteq B$, we have that $\alpha_{\mathfrak{T}}(t_i) \subseteq W_v$ and therefore $\sigma_{\mathfrak{T}}(t_i) \subseteq N^G[W_v] \cap B$. By Claim 2, $|N^G[W_v] \cap B| = O(k^6)$, and we have the required bound on $|\sigma_{\mathfrak{T}}(t)|$. ⌟

Using the bound on the adhesion, it is easy to show that Δ respects $\mathfrak{T}$:

Claim 5. $S_G(\alpha_{\mathfrak{T}}(t_i)) \in \mathfrak{T}$ for every $1 \leq i \leq p$.

Proof. Recall that $\alpha_{\mathfrak{T}}(t_i)$ is disjoint from B and therefore it has to be fully contained in W_v for some $v \in M$: vertices outside every W_v are in B and $N^G(W_v) \subseteq B$. The order of $S_G(\alpha_{\mathfrak{T}}(t_i))$ is exactly $|\sigma_{\mathfrak{T}}(t)|$, which is at most a by Claim 4. As the order of $\mathfrak{T}$ is $m'(k) > a$ and $\alpha_{\mathfrak{T}}(t) \subseteq W_v$, $S_G(W_v) \in \mathfrak{T}$ hold, Lemma 5.6(1) implies that $S_G(\alpha_{\mathfrak{T}}(t_i)) \in \mathfrak{T}$ holds as well. ⌟

The following claim proves the bound on the maximum degree:

Claim 6. There is a constant $d'(k) = O(k^7)$ such that every vertex $v \notin Z$ has degree at most $d'(k)$ in $\tau(s)$,

Proof. Let us observe first that for every $W \in \mathcal{W}$, the graph $\tau(s)$ has no edge between $W \cap B$ and $B \setminus N^G[W]$. To see this, recall that, for every $1 \leq i \leq p$, $\sigma(t_i) = N^G(\alpha_{\mathfrak{T}}(t_i))$, $G[\alpha_{\mathfrak{T}}(t')]$ is connected, and $\alpha_{\mathfrak{T}}(t_i) \cap B = \emptyset$. As $N^G(W) \subseteq B$, it follows that $\alpha_{\mathfrak{T}}(t_i)$ cannot have a neighbor both inside W and outside $N^G[W]$. Therefore, $\sigma(t_i)$ is either a subset $N^G[W]$ or disjoint from W. This means that in the definition of $\tau(s)$, there is no clique that introduces an edge between a vertex in W and a vertex outside $N^G[W]$.

Consider a $u \in B \setminus Z$.

Case 1: $u \in W$ for some $W \in \mathcal{W}$. By our observation above, every neighbor of u in $\tau(t) \setminus Z$ is contained in $N^G[W]$. Therefore, Claim 2 gives a bound of $O(k^6)$ on the degree of u in $\tau(t)$.

Case 2: $u \notin W$ for any $W \in \mathcal{W}$. As $u \notin Z$, this is only possible if the degree of u is at most k in G. Therefore, u is adjacent to at most k components of $G \setminus B$. Each new clique in $\tau(t)$ corresponds to the neighborhood of such a component. Thus u is part of at most k cliques introduced in the definition of $\tau(t)$. The size of each clique can be bounded by the adhesion of Δ, which is at most a by Claim 4. Therefore, k receives at most $k \cdot O(k^6)$ new edges. ⌟

What remains to be proven is the invariance condition. Suppose that $\mathfrak{T}'$ is another tangle of order k in a graph G'. Let f be an isomorphism from $(G, \mathfrak{T})$ to $(G', \mathfrak{T}')$. Let B and B' be the sets computed by the algorithm on $(G, \mathfrak{T})$ and $(G', \mathfrak{T}')$, respectively.

Claim 7. $B' = f(B)$.

Proof. Let $\mathcal{W}$ and $\mathcal{W}'$ be the two collection of sets constructed by the algorithm on $(G, \mathfrak{T})$ and $(G', \mathfrak{T}')$, respectively. Let us observe that $W \in \mathcal{W}$ if and only if $f(W) \in \mathcal{W}'$: the definition of $\mathcal{W}$ depends only on the definition of the sets W_v, which depends only on the tangle $\mathfrak{T}$ and the graph-theoretic properties of G, all of which are preserved by the isomorphism f. Taking into account that the definition of B depends only on the sets in $\mathcal{W}$ and their neighborhoods in G, we can deduce $B' = f(B)$. ⌟

As $B' = f(B)$, for every component of $G \setminus B$, there is a corresponding component of $G' \setminus B'$. Let $C'_1, \ldots, C'_p$ be the components of G', as enumerated by running the

algorithm on $(G', \mathfrak{T}')$, and let s', t'_1, ..., t'_p be the nodes of the constructed star decomposition. Let us define $g(s) = s'$ and let $g(t_i) = t'_j$ such that $f(C_i) = C'_j$.

Claim 8. For all $t \in V(T_{\mathfrak{T}})$ we have $\sigma_{\mathfrak{T}'}(g(t)) = f(\sigma_{\mathfrak{T}}(t))$ and $\alpha_{\mathfrak{T}'}(g(t)) = f(\alpha_{\mathfrak{T}}(t))$.

Proof. The statement immediately follows from the fact that $\alpha_{\mathfrak{T}}(t_i) = C_i$ and $\alpha_{\mathfrak{T}'}(g(t_i)) = f(C_i)$ by definition of G, and hence $\sigma_{\mathfrak{T}}(t_i) = N^G(C_i)$ and $\sigma_{\mathfrak{T}'}(g(t_i)) = N^{G'}(f(C_i)) = f(N^G(C_i))$. ⌟

□

Finally, we can prove Lemma 4.10 by invoking Lemma 6.11 on the tangle defined by the unbreakable set X:

Proof (of Lemma 4.10). Let $c^*(k) = k+3m-2$, $d^*(k) = d'(k)+3m-2$, $\ell^*(k) = \ell'(k)$, $m^*(k) = m'(k)$ for the functions d', ℓ', m' in Lemma 6.11. Let $\mathfrak{T}$ be the tangle of order m defined by the m-unbreakable set X; Lemma 5.1 provides an implementation of the min-cut oracle for $\mathfrak{T}$. Let us call the algorithm of Lemma 6.11 with G, $\mathfrak{T}$, k, ℓ, and m. If it returns a subdivision of K_k in G, then we are done. Otherwise, the algorithm of Lemma 6.11 returns a $\mathfrak{T}$-respecting star decomposition $\Sigma_{\mathfrak{T}} = (T_{\mathfrak{T}}, \alpha_{\mathfrak{T}}, \sigma_{\mathfrak{T}})$ of G. We construct a star decomposition $\Sigma_X = (T_X, \alpha_X, \sigma_X)$ as follows. First, let $T_X = T_{\mathfrak{T}}$ and for the center s of T_X, let $\alpha_X(s) = V(G)$ and $\sigma_X(s) = \varnothing$. For every tip t of T_X, we let $\alpha_X(t) = \alpha_{\mathfrak{T}}(t) \setminus X$ and $\sigma_X(t) = \sigma_{\mathfrak{T}}(t) \cup (X \cap \alpha_{\mathfrak{T}}(t))$. It is straightforward to verify that Σ_X is also a star decomposition of G, and in fact it is star decomposition even for the supergraph $G \cup K[X]$ (since $X \subseteq \beta_X(s)$). As $\tau_{\mathfrak{T}}(s) \setminus X = \tau_X(s) \setminus X$, and $\tau_{\mathfrak{T}}(s)$ contains at most k vertices of degree higher than $d^*(k)$, we have that $\tau_X(s)$ contains at most $k + |X| = c^*(k)$ vertices of degree higher than $d^*(k)$. The bound $< |X|$ on the adhesion and the invariance requirement can be proved the same way as in Lemma 4.9. □

7. PARTIAL DOMINATING SET

The goal of this section is to prove that Partial Dominating Set (find k vertices whose closed neighborhood has maximum size) can be solved in time $f(H,k) \cdot n^{O(1)}$ on graphs excluding H as a topological subgraph. We intend this result as a demonstration of the algorithmic use of the Global Structure Theorem 4.1: it shows that by combining the techniques that work on almost-embeddable and on bounded-degree graphs, we can solve problems on graphs excluding a topological subgraph. We would like to emphasize that all the algorithmic techniques in this section are standard: it is the new structure theorem that allows us to use these standard techniques on a larger class of graphs. We remark that an $f(k) \cdot n^{f(H)}$ algorithm was known for Partial Dominating Set on H-minor free graphs [1], but instead of extending this algorithm, we give here a self-contained presentation of the result on graphs excluding H as a topological subgraph.

We begin by defining a generalization of Partial Dominating Set, which will be convenient for computations on tree decompositions. We extend the problem by introducing a *cost function* $\kappa : V(G) \to \{0,1\}$ and *value function* $\nu : V(G) \to \{0,1\}$; now the goal is to find a set $Z \subseteq V(G)$ with $\kappa(Z) \leq k$ such that $\nu(N^G[Z])$ is maximizied. (As usual, κ and ν are extended to sets by $\nu(Z) = \sum_{v \in Z} \nu(v)$ and $\kappa(Z) = \sum_{v \in Z} \kappa(v)$.) That is, the vertices with $\kappa(v) = 0$ can be used for "free" and the domination of a vertex with $\nu(v) = 0$ does not increase the objective function.

Definition 7.1. *Let G be a graph and $S \subseteq V(G)$ a set of vertices. The* k-profile *of G with respect to S is a function $\pi(z, \kappa, \nu)$, which, for every integer $0 \leq z \leq k$ and functions $\kappa, \nu : V(G) \to \{0,1\}$ that have value 1 on $V(G) \setminus S$, gives the maximum of $\nu(N^G[Z])$ taken over every $Z \subseteq V(G)$ with $\kappa(Z) \leq z$.*

That is, the k-profile with respect to S is described by $(k+1) \cdot 2^{|S|} \cdot 2^{|S|}$ integers. Observe that if the k-profile with respect to S is known, then it is easy to deterimine the k-profile with respect to some $S' \subseteq S$.

First we show that the k-profile can be computed in a bottom-up manner on a tree decomposition if every bag is small, that is, the decomposition has bounded width. Then we use a standard layering argument to compute the k-profile on almost-embeddable torsos by reducing it to the bounded-treewidth case. For this reduction, we need the fact that almost-embeddable graphs have bounded *local treewidth:*

Theorem 7.2 ([10]). *For every $p, q, r \in \mathbb{N}$, there is a constant $\lambda > 0$ such that the following holds. Let G be a minor of a $(p,q,r,0)$-almost embeddable graph, let $x \in V(G)$, and let $N_d(\{x\}) \subseteq V(G)$ be the set of vertices at distance at most d from x. Then $G[N_d(\{x\})]$ has treewidth at most $\lambda \cdot d$ for every $d \geq 0$.*

Finally, we compute the k-profile on almost bounded-degree torsos by using a standard random coloring technique.

Lemma 7.3. *Let (T, σ, α) be a tree decomposition of a graph G and t a node of T. Suppose that, for every child t' of t, the k-profile of $G[\gamma(t')]$ with respect to $\sigma(t')$ is known. Then the k-profile of $G[\gamma(t)]$ with respect to $\sigma(t)$ can be computed*

(1) in time $f(s) \cdot n^{O(1)}$ if $|\beta(t)| \leq s$ and $|N^T(t)| \leq 2$.
(2) in time $f(w) \cdot n^{O(1)}$ if a tree decomposition of $\tau(t)$ having width w is given.
(3) in time $f(k,p,q,r,s,a) \cdot n^{O(1)}$ if $|\sigma(t)| \leq a$ and a set P of size at most s is given such that $\tau(t) \setminus P$ is almost $(p,q,r,0)$-embeddable.
(4) in time $f(k,c,d,a) \cdot n^{O(1)}$ if $|\sigma(t)| \leq a$ and all but at most c vertices have degree at most d in $\tau(t)$.

Theorem 1.2 follows immediately by putting together Corollary 4.4 and Lemma 7.3(3–4): in a bottom-up order, for every node t of the decomposition given by Corollary 4.4, we can compute the k-profile of $G[\gamma(t)]$ with respect to $\sigma(t)$, which gives us the value of the optimum solution of Partial Dominating Set.

Recall that a graph is *d-degenerate* if every subgraph has a vertex of degree at most d. A classical result of Mader [19] shows that every graph excluding H as a topological subgraph is d_H-degenerate for some constant d_H depending on H, thus it is a natural question whether Theorem 1.2 can be generalized to the more general class of d-degenerate graphs. However, Partial Dominating Set is W[1]-hard parameterized by k and d on d-degenerate graphs. To see this, note that Maximum Independent Set, parameterized by the size k of the solution, is W[1]-hard even on regular graphs. Let G be an r-regular graph ($r \geq 3$) and let us subdivide every edge by a new vertex. It is not difficult to see that G

has an independent set of size k if and only if the new graph G' has a set of k vertices whose closed neighborhood has size $(r+1)k$. As G' is 2-degenerate, an $f(k,d)\cdot n^{O(1)}$ time algorithm for Partial Dominating Set on d-degenerate graphs would imply an $f(k)\cdot n^{O(1)}$ time algorithm for Maximum Independent Set. Thus the fixed-parameter tractability of Partial Independent Set on graph excluding H as a topological subgraph is not simply a consequence of the sparsity/degeneracy of such graphs, but essentially depends on the structural properties of this class of graphs.

8. COMPUTING INVARIANT TREELIKE DECOMPOSITIONS

In this section, we relax the notion of tree decomposition to the more liberal notion of treelike decomposition. The reason is that we want to make our decompositions invariant under automorphisms of the underlying graph, and this is not possible for tree decompositions. Treelike decompositions are based on the axiomatisation of tree decompositions by (TD.1)–(TD.5). From now on, a *decomposition* of a graph is a triple $\Delta = (D,\sigma,\alpha)$, where D is a digraph and $\sigma,\alpha : V(D) \mapsto 2^{V(G)}$. For every $t \in V(D)$, we define sets $\gamma(t),\beta(t) \subseteq V(G)$ and a graph $\tau(t)$ as in (3.4), (3.5), and (3.6). The *width* and *adhesion* of a decomposition are defined, as for tree decompositions, to be the maximum size of the bags minus one and the maximum size of the separators, respectively. Two nodes $t,u \in V(D)$ are Δ-*equivalent* (we write $t \bowtie u$) if $\sigma(t)=\sigma(u)$ and $\alpha(t)=\alpha(u)$. Note that $t \bowtie u$ implies $\gamma(t)=\gamma(u)$, but not $\beta(t)=\beta(u)$ or $\tau(t)=\tau(u)$. We will occasionally work with several decompositions at the same time, and in such situations may use an index $^\Delta$, as for example in $\sigma^\Delta(t)$ or $t \bowtie^\Delta u$, to indicate which decomposition we are referring to. However, we usually prefer implicit naming conventions such as the following: If we have a decomposition $\Delta' = (D',\sigma',\alpha')$, then we will denote $\gamma^{\Delta'}(t)$ by $\gamma'(t)$, $\beta^{\Delta'}(t)$ by $\beta'(t)$, et cetera.

Definition 8.1. *A* treelike decomposition *of a graph G is a decomposition $\Delta = (D,\sigma,\alpha)$ of G that satisfies the following axioms:*

(TL.1) *D is acyclic.*

(TL.2) *For all $t \in V(D)$ it holds that $\alpha(t)\cap\sigma(t)=\emptyset$ and $N^G(\alpha(t)) \subseteq \sigma(t)$.*

(TL.3) *For all $t \in V(D)$ and $u \in N^D(t)$ it holds that $\alpha(u) \subseteq \alpha(t)$ and $\gamma(u)\subseteq\gamma(t)$.*

(TL.4) *For all $t \in V(D)$ and $u_1,u_2 \in N^D(t)$, either $u_1 \bowtie u_2$ or $\gamma(u_1)\cap\gamma(u_2)=\sigma(u_1)\cap\sigma(u_2)$.*

(TL.5) *For every connected component A of G there is a $t \in V(D)$ with $\sigma(t)=\emptyset$ and $\alpha(t)=V(A)$.*

Note that (TD.2) coincides with (TL.2) and (TD.3) coincides with (TL.3). Moreover, (TD.1) implies (TL.1) and (TD.4) implies (TL.4). For connected graphs G, (TD.5) coincides with (TL.5), and thus every tree decomposition of a connected graph is a treelike decomposition. For disconnected graphs, this is not necessarily the case, but it can be shown that from every treelike decomposition one can construct a tree decomposition with the same torsos. (See [9] for details.)

Figure 8.1(a) shows the cycle C_5. Figure 8.1(b) shows a tree decomposition (T,β) of C_5 of width 2. Note that this tree decomposition is not invariant under automorphisms of C_5, in the sense that there is an automorphism f of C_5 for which we cannot find an automorphism g of T such that for all $t \in V(T)$ we have $f(\beta(t)) = \beta(g(t))$. It is easy to see that there is no tree decomposition of C_5 of width 2 that is invariant under automorphisms.

Figure 8.1(b) shows a treelike decomposition (D',σ',α') of C_5 of width 2. Actually, the sets displayed in the nodes are the bags, but we can easily compute the separators and components using (3.1) and (3.3). For instance, for the grey node t with bag $\beta'(t)=\{1,3,5\}$ we have $\sigma'(t)=\{1,3\}$ and $\alpha'(t)=\{4,5\}$. For the sake of completeness, we observe that $\gamma'(t)=\{1,3,4,5\}$ and $\tau'(t)=K[\{1,3,5\}]$.

Note that the "subdecomposition" induced by the four grey nodes is precisely the tree decomposition shown in Figure 8.1(b). The treelike decomposition contains many other tree decompositions of C_5; actually, it contains all images of the decomposition shown in Figure 8.1(b) under automorphisms of C_5. And indeed, the treelike decomposition is invariant under automorphisms.

This example illustrates how treelike decompositions can be made "invariant." However, the automorphism invariance of the example is not sufficient for our purposes, we need a more general notion of invariance that involves decompositions of more than one graph.

Definition 8.2. *A* decomposition mapping *for a class $\mathcal{C}$ of graphs is a mapping Δ that associates with each $G \in \mathcal{C}$ a decomposition $\Delta_G = (D_G,\sigma_G,\alpha_G)$ of G.*

Δ is invariant *if for all isomorphic graphs $G,G' \in \mathcal{C}$ and all isomorphisms f from G to G' there is an isomorphism g from D_G to $D_{G'}$ such that for all $t \in V(D_G)$ we have $\sigma_{G'}(g(t)) = f(\sigma_G(t))$ and $\alpha_{G'}(g(t)) = f(\alpha_G(t))$.*

We need some additional terminology about decomposition mappings: We say that a decomposition mapping Δ for a class $\mathcal{C}$ is *treelike* if for all $G \in \mathcal{C}$ the decomposition Δ_G is treelike, and it has *adhesion at most a* if for all $G \in \mathcal{C}$ the adhesion of Δ_G is at most a. We say that a class $\mathcal{C}$ *admits polynomial time computable invariant treelike decompositions over $\mathcal{A}$ (of adhesion at most a)* if there is a polynomial time computable invariant treelike decomposition mapping for $\mathcal{C}$ over $\mathcal{A}$ (of adhesion a). Let us remark that the decomposition schemes of [9] yield polynomial time computable invariant decomposition mappings.

The main result of the section is the following:

Theorem 8.3 (Invariant Decomposition Theorem). *For every graph H there are constants $a,b,c,d,e \in \mathbb{N}$ and a polynomial time computable invariant treelike decomposition mapping Δ of adhesion at most a for the class of graphs G with $H \not\leq_T G$ such that for every G with $\Delta_G = (D,\sigma,\alpha)$ and every $t \in V(D)$ one of the following three conditions is satisfied:*

(i) $|\beta(s)| \le b$
(ii) At most c vertices of $\tau(t)$ have degree greater than d.
(iii) $K_e \not\leq \tau(t)$.

Proof. We let $k := |H|$. We choose $c = c^*(k)$, $d = d^*(k)$, $\ell = \ell^*(k)$, and $m = m^*(k)$ according to Lemma 4.10 and $e = e^*(\ell,m)$ according to Lemma 4.9. We let $a := 3m-3$ and $b := 4m-3$..

Let G be a graph with $H \not\leq_T G$. We shall define a decomposition $\Delta_G = (D,\sigma,\alpha)$ of G of adhesion at most a such that every node t satisfies one of (i)–(iii). Then we will argue that

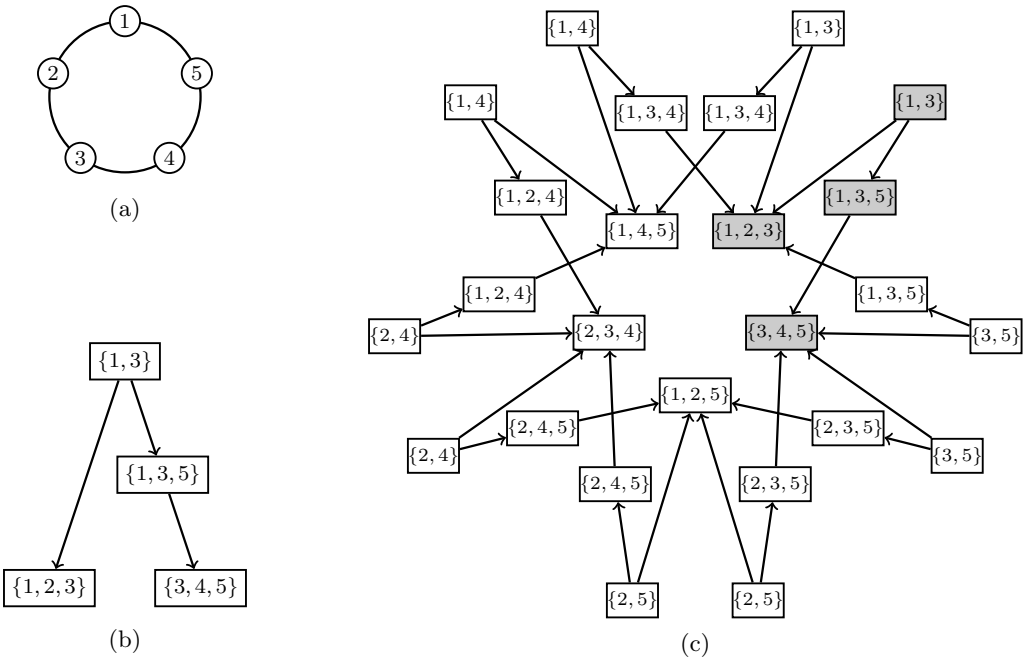

Figure 8.1: (a) The cycle C_5 with (b) a tree decomposition and (c) an automorphism-invariant treelike decomposition

the decomposition $G \mapsto \Delta_G$ is polynomial time computable and invariant.

There will be three kinds of nodes in $V(D)$: b-nodes ("bounded nodes"), d-nodes ("bounded degree nodes"), and e-nodes ("excluded minor nodes"). All nodes are triples $t = (A_t, X_t, Y_t)$ satisfying the following conditions:

(A) A_t is a connected induced subgraph of G with $|N^G(A_t)| \leq a$. To simplify the notation, in the following we let $C_t := G\big[N^G[A_t]\big]$.
(B) $X_t \subseteq V(C_t)$ such that $N^G(A_t) \subset X_t$ and $|X_t| = \min\{a+1, |C_t|\}$.
(C) $Y_t \subseteq V(C_t)$ such that $|Y_t| < m$. (Actually, Y_t will be empty for d-nodes and e-nodes.)

Let us call such triples "nodes" and let U be the set of all nodes (the actual nodes of D will form a subset of U). For every node $t \in U$ we let $\alpha(t) := V(A_t)$, $\sigma(A_t) := N^G(A_t)$, and $\gamma(t) := V(C_t)$.

(D) A *b-node* is a node $t \in U$ such that for every connected component A of $C_t \setminus Y_t$ it holds that $|(V(A) \cap X_t) \cup Y_t| < |X_t|$.

Let V_b be the set of all b-nodes. Let U_b be the set of all nodes $t \in U$ for which there exists a $Y \subseteq V(C_t)$ of size $|Y| < m$ such that for every connected component of A of $C_t \setminus Y$ it holds that $|(V(A) \cap X_t) \cup Y| < |X_t|$. Note that $V_b \subseteq U_b$ and that that for every $t \in U \setminus U_b$ the set X_t is m-unbreakable in C_t.

(E) An *e-node* is a node $t \in U \setminus U_b$ such that $Y_t = \varnothing$ and the algorithm of Lemma 4.9 on C_t, ℓ, m, and X_t returns star decomposition $\Sigma_t := \Sigma_{X_t}$ of $C_t \cup K[X_t]$.
(F) A *d-node* is a node $t \in U \setminus U_b$ such that $Y_t = \varnothing$ and the algorithm of Lemma 4.9 on C_t, ℓ, m, and X_t returns an image I of K_ℓ in C_t that is m-attached to X_t. In this case, the algorithm of Lemma 4.10 applied to C_t, k, the set X_t, and the image I computes a star decomposition $\Sigma_t := \Sigma_{X_t}$ of $C_t \cup K[X_t]$ (since $K_k \not\leq_T C_t$ by assumption).

Let V_d and V_e be the sets of d-nodes and e-nodes, respectively. Note that the three sets V_b, V_d, V_e are mutually disjoint. We let $V(D) := V_b \cup V_d \cup V_e$.

Claim 1. Let A be a (nonempty) connected induced subgraph of G with $|N^G(A)| \leq a$. Then there is a node $t \in V(D)$ such that $A_t = A$.

Proof. Let $C := G\big[N^G[A]\big]$, and choose an arbitrary $X \subseteq V(C)$ such that $N^G(A) \subset X$ and $|X| = \min\{a+1, |C|\}$. Clearly, such a set X exists, because $A \neq \varnothing$ and $|N^G(A)| \leq a$.

If there is a set $Y \subseteq V(G)$ such that $|Y| < m$ and for every connected component A' of $C \setminus Y$ it holds that $|(V(A') \cap X) \cup Y| < |X|$, then $(A, X, Y) \in V_b$.

Suppose there is no such set Y. Then $(A, X, \varnothing) \notin U_b$ and thus $(A, X, \varnothing) \in V_e \cup V_d$. ⌟

By (E) and (F), for all $t \in V_e \cup V_d$ we have a star decomposition $\Sigma_t =: (T_t, \sigma_t, \alpha_t)$. Let s_t be the center of T_t. To define the edge relation $E(D)$, for every node $t \in V(D)$ we define the set $N^D(t)$ of its children in D.

(G) For $t \in V_b$, we let $N^D(t)$ be the set of all $u \in V(D)$ such that A_u is a connected component of $C_t \setminus (X_t \cup Y_t)$.
(H) For $t \in V_d \cup V_e$, we let $N^D(t)$ be the set of all $u \in V(D)$ such that A_u is a connected component of $C_t \setminus \beta_t(s_t)$.

This completes the definition of the decomposition $\Delta_G = (D, \sigma, \alpha)$.

Claim 2. Δ_G is a treelike decomposition of G.

Proof. It follows immediately from the definitions of σ and α that Δ_G satisfies (TL.2).

To verify (TL.3), let $tu \in E(D)$. We have $X_t \subseteq \beta_t(s_t)$ (either by (G) or by the statements of Lemmas 4.9 and 4.10). Therefore, by (G) and (H) we have

$$\begin{aligned}\alpha(u) = V(A_u) &\subseteq V(C_t) \setminus \beta_t(s_t) \\ &\subseteq V(C_t) \setminus X_t \subset V(C_t) \setminus N^G(A_t) = \alpha(t).\end{aligned} \tag{8.1}$$

Moreover, if $t \in V_b$, then we have $N^{C_t}(A_u) \subseteq X_t \cup Y_t$. Since every vertex of C_t with a neighbor outside C_t is in $N^G(A_t) \subseteq X_t$ and we have $V(A_u) \cap X_t = \varnothing$, this implies $N^G(A_u) \subseteq X_t \cup Y_t$. Hence $\gamma(u) = V(A_u) \cup N^G(A_u) \subseteq V(C_t) = \gamma(t)$. If $t \in V_e \cup V_b$ then we have $N^{C_t}(A_u) \subseteq \beta_t(s_t)$. Again, every vertex of C_t with a neighbor outside C_t is in $N^G(A_t) \subseteq X_t \subseteq \beta_t(s_t)$ and $V(A_u) \cap \beta_t(s_t) = \varnothing$. Hence $\gamma(u) = V(A_u) \cup N^G(A_u) \subseteq V(C_t) = \gamma(t)$.

Note that in (8.1) we proved that for all edges $tu \in E(D)$ the inclusion $\alpha(u) \subset \alpha(t)$ is strict. This implies that D is acyclic, that is, (TL.1).

To verify (TL.4), let $t \in V(D)$ and $u_1, u_2 \in N^D(t)$. For $i = 1, 2$, we let $C_i := C_{u_i}$ and $X_i := X_{u_i}$ and $A_i := A^t_{u_i}$.

Case 1: $t \in V_b$.
Then by (G), both A_1 and A_2 are connected components of $C_t \setminus (X_t \cup Y_t)$. Hence either $A_1 = A_2$ or $A_1 \cap A_2 = \varnothing$. If $A_1 = A_2$ then $\alpha(u_1) = V(A_1) = V(A_2) = \alpha(u_2)$ and $\sigma(u_1) = N^G(A_1) = N^G(A_2) = \sigma(u_2)$ and thus $u_1 \backsimeq u_2$. Suppose that $A_1 \cap A_2 = \varnothing$. Note that we also have $V(A_1) \cap N^G(A_2) \subseteq A_1 \cap (X_t \cup Y_t) = \varnothing$ and, symmetrically, $V(A_2) \cap N^G(A_1) = \varnothing$. This implies $\gamma(u_1) \cap \gamma(u_2) = V(C_1) \cap V(C_2) = N^G(A_1) \cap N^G(A_2) = \sigma(u_1) \cap \sigma(u_2)$.

Case 2: $t \in V_e \cup V_d$.
Then both A_1 and A_2 are connected components of $C_t \setminus \beta_t(s_t)$, and we can argue as in Case 1.

To verify (TL.5), let A be a connected component of G. Then $N^G(A) = \varnothing$, and by Claim 1 there is a $t \in V(D)$ such that $C_t = A$. For each such t, we have $\sigma(t) = N^G(A) = \varnothing$ and $\alpha(t) = V(A)$. ⌟

Claim 3. Let $t \in V(D)$.

(1) If $t \in V_b$ then $\beta(t) = X_t \cup Y_t$.
(2) If $t \in V_e \cup V_d$ then $\beta(t) = \beta_t(s_t)$ and $\tau(t) \subseteq \tau_t(s_t)$.

Proof. Recall that $\beta(t) = \gamma(t) \setminus \bigcup_{u \in N^D(t)} \alpha(u) = V(C_t) \setminus \bigcup_{u \in N^D(t)} V(A_u)$.

To prove (1), suppose that $t \in V_b$. It follows from (G) that for all $u \in N^D(t)$ we have $V(A_u) \subseteq V(C_t) \setminus (X_t \cup Y_t)$. Hence $(X_t \cup Y_t) \subseteq \beta(t)$. To prove the converse inclusion, we shall prove that for every connected component A of $C_t \setminus (X_t \cup Y_t)$ there is a $u \in N^D(t)$ with $A_u = A$. By (G) and Claim 1, it suffices to prove that $|N^G(A)| \leq a$. But this follows from the definition of b-nodes in (D) (we used a similar argument in the proof of Claim 1).

To prove (2), let $t \in V_e \cup V_d$. By (H), for all $u \in N^D(t)$ we have $V(A_u) \subseteq V(C_t) \setminus \beta_t(s_t)$. Hence $\beta_t(s_t) \subseteq \beta(t)$. For the converse inclusion, let A be a connected component of $C_t \setminus \beta_t(s_t)$. Then there is a tip x of T_t such that A is a connected component of $C_t[\alpha_t(x)] = G[\alpha_t(x)]$. We have $N^G(A) \subseteq \sigma_t(x)$, and as the adhesion of Σ_t is $< |X_t|$, we have $|N^G(A)| \leq |\sigma_t(x)| < |X_t| = a + 1$. Thus by Claim 1 and (H), there is a $u \in N^D(t)$ with $A_u = A$.

It remains to prove that $\tau(t) \subseteq \tau_t(s_t)$. First, note that for all $u \in N^D(t)$ there is an $x \in N^{T_t}(s_t)$ such that $\sigma(u) \subseteq \sigma_t(x)$. Furthermore, Σ_t is a decomposition of $C_t \cup K[X_t]$, thus $\sigma(t) \subseteq X_t$ is a clique in $\tau_t(s_t)$. Let us remark that $\tau_t(s_t) \subseteq \tau(t)$ is not necessarily true: X_t is a proper superset of $\sigma(t)$, thus X_t is a clique $\tau_t(s_t)$, but it is not necessarily a clique in $\tau(t)$. ⌟

It follows from (A) that the adhesion of Δ_G is at most a. By Claim 3(1), every $t \in V_b$ satisfies (i). By Claim 3(2) and Lemmas 4.10 and 4.9, every $t \in V_d$ satisfies (ii) and every $t \in V_e$ satisfies (iii).

It it easy to see that the decomposition mapping Δ is polynomial time computable. Indeed, note first that the set U has size $O(n^{a+1+3m-2+m-1})$ (here we use n^{a+1} as an upper bound for the number of connected induced subgraphs A of G with $N^G(A)| \leq a$) and that the set is polynomial time computable. Remember that the parameters a, m et cetera are all treated as constants depending only on H. The subset U_b is also polynomial time computable, because to decide whether $t \in U_b$ we can go through all subsets $Y \subseteq V(C_t)$ of size less than m and see if the condition is satisfied. Now it follows from Lemmas 4.9 and 4.10 that the sets V_e and V_d are polynomial time computable. Hence $V(D)$ is polynomial time computable. Since for nodes $t \in V_e \cup V_d$ the star decomposition Σ_t is polynomial time computable (again by Lemmas 4.9 and 4.10), the edge relation $E(D)$ is polynomial time computable as well. Since the mappings σ and α are almost trivially polynomial time computable, this shows that Δ is polynomial time computable.

It remains to prove that Δ is invariant. To prove this, we take isomorphic graphs G, G' with $H \not\leq G, G'$ and let f be an isomorphism from G to G'. Let $\Delta_G = (D, \sigma, \alpha)$ and $\Delta_{G'} = (D', \sigma', \alpha')$. We define the sets U, U_b, V_b, V_e, V_d for G as above and let $U', U'_b, V'_b, V'_e, V'_d$ be the corresponding sets for G'. We denote the constituents of a node $t' \in U'$ by $(A'_{t'}, X'_{t'}, Y'_{t'})$ and let $C'_{t'} := G'\big[N^{G'}[A'_t]\big]$. For $t' \in V_e \cup V_d$ we denote the star decomposition of $C'_{t'}$ (obtained as above) by $\Sigma'_{t'}$. The isomorphism f has a natural extension of subsets of $V(G)$, tuples of subsets, and similar objects defined in terms of $V(G)$. We denote this extension by f^*. As f is an isomorphism, we obviously have $f^*(U) = U'$, $f^*(U_b) = U'_b$, and $f^*(V_b) = V'_b$. Moreover, for every $t \in U$ we have $f^*(A_t) = A'_{f^*(t)}$, $f^*(C_t) = C'_{f^*(t)}$, et cetera. It follows from the invariance conditions of Lemmas 4.9 and 4.10 that $f^*(V_e) = V'_e$ and $f^*(V_d) = V'_d$ and that for every $t \in V_e \cup V_d$ there is an isomorphism g_t from T_t to T'_t such that $f^*(\sigma_t(x)) = \sigma_{f^*(t)}(g_t(x))$ and $f^*(\alpha_t(x)) = \alpha_{f^*(t)}(g_t(x))$ for all $x \in V(T_t)$. But this implies that $N^D(f^*(t)) = \{f^*(u) \mid u \in N^D(t)\}$. As $f^*(C_t) = C'_{f^*(t)}$ and $f^*(X_t) = X'_{f^*(t)}$ and $f^*(Y_t) = Y'_{f^*(t)}$, we also have $N^D(f^*(t)) = \{f^*(u) \mid u \in N^D(t)\}$ for all $t \in V_b$. Hence the restriction of f^* to $V(D)$ is an isomorphism from D to D'. As $f^*(A_t) = A'_{f^*(t)}$ for all $t \in V(D)$, we have $f(\alpha(t)) = \alpha'(f^*(t))$ and thus $f(\sigma(t)) = f(N^G(\alpha(t))) = N^{G'}(f(\alpha(t))) = N^{G'}(\alpha'(f^*(t))) = \sigma'(f^*(t))$. This proves that Δ is invariant. □

9. CANONIZATION

A *canonisation mapping* $\mathfrak{c}$ for a class $\mathcal{C}$ of graphs is a mapping that associates with each graph $G \in \mathcal{C}$ a graph $\mathfrak{c}(G) \cong G$ such that for all $G, H \in \mathcal{C}$ we have $G \cong H \iff \mathfrak{c}(G) = \mathfrak{c}(H)$. That is, $\mathfrak{c}(G)$ and $\mathfrak{c}(H)$ are not only isomorphic, but they are actually the same graph on the same set of vertices. Thus the isomorphism of G and H can be tested simply by comparing $\mathfrak{c}(G)$ and $\mathfrak{c}(H)$. A *canonisation algorithm* com-

putes a canonisation mapping. Without loss of generality we may always assume a canonisation mapping $\mathfrak{c}$ to map graphs G to graphs $\mathfrak{c}(G)$ with vertex set $V(\mathfrak{c}(G)) = [n]$, where $n := |G|$. We say that a class $\mathcal{C}$ of graphs *admits polynomial time canonisation* if there is a polynomial time algorithm that computes a canonisation mapping for $\mathcal{C}$.

FACT 9.1 (BABAI AND LUKS [2]). *For every $d \in \mathbb{N}$ the class of all graphs of maximum degree at most d admits polynomial time canonisation.*

FACT 9.2 (PONOMARENKO [22]). *For every graph H the class of all graphs excluding H as a minor admits polynomial time canonisation.*

Our goal in this section is to prove a "Lifting Lemma" that allows us to lift a canonisation from the torsos of a treelike decomposition of a graph to the whole graph. To be able to prove such a lemma, we need to work with more general structures than graphs and a stronger notion of canonisation.

We often denote tuples $(v_1, \dots, v_k)$ by $\vec{v}$. For $\vec{v} = (v_1, \dots, v_k)$, by $\tilde{v}$ we denote the set $\{v_1, \dots, v_k\}$. A *vocabulary* is a finite set of relation symbols, each of which has a prescribed arity in $\mathbb{N}$. (Note that we admit 0-ary relation symbols. For every set S the set S^0 just consists of the empty tuple.) Let λ be a vocabulary. A *weighted λ-structure* A consists of a *universe* (or *vertex set*) $V(A)$ and for each k-ary relation symbol $R \in \lambda$ a mapping $R^A : V(A)^k \to \mathbb{N}$. A (plain) λ-structure is a weighted λ-structure A with $\text{range}(R^A) \subseteq \{0,1\}$ for all $R \in \lambda$. We usually identify a function $R^A : V(A)^k \to \{0,1\}$ with the relation $R(A) := \{\vec{v} \in V(A)^k \mid R^A(\vec{v}) = 1\}$ and view a plain structure as a finite set (the universe) together with a collection of relations on this universe. For example, graphs and digraphs may be viewed as plain $\{E\}$-structures, where E is a binary relation symbol. Graphs with multiple edges may be viewed as weighted $\{E\}$-structures.

Let λ, μ be vocabularies with $\lambda \subseteq \mu$, and let A be a weighted λ-structure and B a weighted μ-structure. Then A is the *λ-restriction* of B if $V(A) = V(B)$ and $R^A = R^B$ for all symbols $R \in \lambda$. Conversely, B is a *μ-expansion* of A if A is the λ-restriction of B. For every $W \subseteq V(A)$, we define the *induced substructure* $A[W]$ to be the weighted λ-structure with universe $V(A[W]) := W$, relations $R^{A[W]} := R^A \restriction_{W^k}$ for all k-ary $R \in \lambda$. We let $A \setminus W := A[V(A) \setminus W]$. If f is a mapping with domain $V(A)$, we let $f(A)$ be the weighted λ-structure with universe $V(f(A)) := f(V(A))$ and mappings $R^{f(A)}$ defined by $R^{f(A)}(f(\bar{a})) := R^A(\bar{a})$. If A and B are weighted λ-structures such that for all k-ary $R \in \lambda$ and all $\vec{v} \in V(A)^k \cap V(B)^k$ we have $R^A(\vec{v}) = R^B(\vec{v})$, then we define the union $A \cup B$ to be the weighted λ-structure with $V(A \cup B) := V(A) \cup V(B)$ and

$$R^{A \cup B}(\vec{a}) := \begin{cases} R^A(\vec{a}) & \text{if } \vec{a} \in V(A)^k, \\ R^B(\vec{a}) & \text{if } \vec{a} \in V(B)^k, \\ 0 & \text{otherwise.} \end{cases}$$

for all k-ary relation symbols $R \in \lambda$ and $\vec{a} \in V(A \cup B)^k$.

The *Gaifman graph* of a weighted λ-structure A is the graph G_A with vertex set $V(G_A) := V(A)$ and edge set

$$E(G_A) := \Big\{ vw \in \binom{V(A)}{2} \Big| \exists k\text{-ary } R \in \lambda, \vec{v} \in V(A)^k \text{ with } R^A(\vec{v}) > 0 \text{ and } v, w \in \tilde{v} \Big\}.$$

An *isomorphism* from a weighted λ-structure A to a weighted λ-structure B is a bijective mapping $f : V(A) \to V(B)$ such that for all k-ary $R \in \lambda$ and all $\vec{v} \in V(A)^k$ we have $R^A(\vec{v}) = R^B(f(\vec{v}))$. Canonisation mappings and algorithms for weighted structures are defined in the obvious way. We say that a class $\mathcal{C}$ of graphs *admits polynomial time strong canonisation* if for every vocabulary λ there is a polynomial time computable canonisation mapping for the class of all weighted λ-structures with Gaifman graph in $\mathcal{C}$. The following statements can be derived from Facts 9.1 and 9.2 by a simple gadget construction.

LEMMA 9.3. *For all $d \in \mathbb{N}$ the class of all graphs of maximum degree at most d admits polynomial time strong canonisation.*

LEMMA 9.4. *For every graph H, the class of graphs G with $H \not\leq G$ admits polynomial time strong canonisation.*

COROLLARY 9.5. *For all $c, d \in \mathbb{N}$ the class of all graphs G such that at most c vertices of G have degree greater than d admits polynomial time strong canonisation and $k \in \mathbb{N}$.*

The main result of the section is the following lemma:

LEMMA 9.6 (LIFTING LEMMA). *Let $\mathcal{A}, \mathcal{C}$ be a classes of graphs and $a \in \mathbb{N}$. Suppose that $\mathcal{A}$ admits polynomial time strong canonisation and that $\mathcal{C}$ admits polynomial time computable invariant treelike decompositions over $\mathcal{A}$ of adhesion a.*

Then $\mathcal{C}$ admits polynomial time strong canonisation.

Before we prove the lemma, we define the *lexicographical order* $\leq^\lambda_{\text{lex}}$ on weighted λ-structures A with $V(A) \subseteq \mathbb{N}$. Let $\lambda = \{R_1, \dots, R_m\}$, where R_i is k_i-ary. The order $\leq^\lambda_{\text{lex}}$ actually not only depends on the set λ, but on the order in which the relations are listed. Hence we fix this order. We first review the lexicographical order on tuples and sets of integers:

- For tuples $\vec{x} = (x_1, \dots, x_k) \in \mathbb{N}^k$, $\vec{y} = (y_1, \dots, y_\ell) \in \mathbb{N}^\ell$ we let $\vec{x} <_{\text{lex}} \vec{y}$ if and only if either there exists an $i \leq \min\{k, \ell\}$ such that $x_i < y_i$ and $x_j = y_j$ for $1 \leq j < i$ or $k < \ell$ and $x_i = y_i$ for all $i \leq k$.
- For sets $X, Y \subseteq \mathbb{N}$ we let $X <_{\text{lex}} Y$ if and only if there exists an $i \in Y \setminus X$ such that for all $j < i$ it holds that $j \in X \iff j \in Y$.

Now let A, B be weighted λ-structures with $V(A), V(B) \subseteq \mathbb{N}$. Then we let $A <^\lambda_{\text{lex}} B$ if one of the following conditions is satisfied:

- $V(A) <_{\text{lex}} V(B)$. Note that if both $V(A)$ and $V(B)$ are initial segments of the positive integers then this just means $|A| < |B|$.
- $V(A) = V(B) =: V$ and there is an $i \in [m]$ and a tuple $\vec{a} \in V^{k_i}$ such that $R^A_j = R^B_j$ for $1 \leq j < i$, and $R^A_i(\vec{a}) < R^B_i(\vec{a})$, and $R^A_i(\vec{b}) = R^B_i(\vec{b})$ for all $\vec{b} \in V^{k_i}$ with $\vec{b} <_{\text{lex}} \vec{a}$.

We let $A \leq^\lambda_{\text{lex}} B$ if $A <^\lambda_{\text{lex}} B$ or $A = B$. Note that $\leq^\lambda_{\text{lex}}$ is indeed a linear order on the class of weighted λ-structures whose universe is a set of natural numbers and that given A, B, it can be decided in polynomial time whether $A \leq^\lambda_{\text{lex}} B$.

As another small piece of terminology, we say that an *enumeration* of a finite set S is a tuple $(x_1, \dots, x_k)$ such that $k = |S|$ and $S = \{x_1, \dots, x_k\}$.

Proof (of the Lifting Lemma 9.6). We shall describe a polynomial time computable a canonisation mapping $\mathfrak{c}$ for the class of all weighted λ-structures with Gaifman graph in $\mathcal{C}$.

Let $P_1, \dots, P_a, Q_1, \dots, Q_a \notin \lambda$ be fresh relation symbols, where P_i and Q_i are i-ary for all $i \in [a]$. Let $\mu := \lambda \cup \{P_1, \dots, P_a, Q_1, \dots, Q_a\}$. Let $\mathfrak{a}$ be a polynomial time computable canonisation mapping on the class of all weighted μ-structures whose Gaifman graph is in $\mathcal{A}$. Such a mapping $\mathfrak{a}$ exists by the assumption that $\mathcal{A}$ admits polynomial time strong canonisation. Let Δ be a polynomial time computable invariant treelike decomposition mapping for $\mathcal{C}$ over $\mathcal{A}$ of adhesion at most a.

To explain our canonisation mapping $\mathfrak{c}$, let us fix a λ-structure C with Gaifman graph $G_C \in \mathcal{C}$. Without loss of generality we may assume that G_C is connected. Let $\Delta_{G_C} = (D, \sigma, \alpha)$. For every $t \in V(D)$ we let $C_t := C[\gamma(t)]$ and $s_t := |\sigma(t)|$. Note that $0 \leq s_t \leq a$. By induction on D, starting from the leaves, for every node $t \in V(D)$ and every enumeration $\vec{x}$ of $\sigma(t)$ we define a copy $C^*_{t,\vec{x}}$ of C_t and a mapping $g_{t,\vec{x}} : \sigma(t) \to V(C^*_{t,\vec{x}})$ with the following properties:

(A) $V(C^*_{t,\vec{x}})$ is an initial segment of the positive integers.
(B) There is an isomorphism f from C_t to $C^*_{t,\vec{x}}$ such that $g_{t,\vec{x}} \subseteq f$.

Let $t \in V(D)$, and let $\vec{x} = (x_1, \dots, x_{s_t})$ be an enumeration of $\sigma(t)$. Let $u_1, \dots, u_m$ be the children of t in D. For every $i \in [m]$, let $C_i := C_{u_i}$ and $n_i := |\alpha(u_i)|$ and $s_i := s_{u_i}$. Note that $|C_i| = n_i + s_i$. For all $i, j \in [m]$, let $i \backsimeq j :\Leftrightarrow u_i \backsimeq^{\Delta_{G_C}} u_j$. For every $i \in [m]$ and every tuple $\vec{y}$ that enumerates $\sigma(u_i)$, let $C^*_{i,\vec{y}} := C^*_{u_i,\vec{y}}$ and $g_{i,\vec{y}} := g_{u_i,\vec{y}}$. Then $C^*_{i,\vec{y}}$ and $g_{i,\vec{y}}$ satisfy (A) and (B).

Let $\mathcal{Y}$ be the set of all $\vec{y} \in \beta(t)^{\leq a}$ that enumerate $\sigma(u_i)$ for some $i \in [m]$. For each $\vec{y} \in \mathcal{Y}$, let $M_{\vec{y}}$ be the set of all $i \in [m]$ with $\sigma(u_i) = \tilde{y}$. Let $\preceq_{\vec{y}}$ be a linear order on $M_{\vec{y}}$ such that for all $i, j \in M_{\vec{y}}$:

(C) If $C^*_{i,\vec{y}} <^{\lambda}_{\text{lex}} C^*_{j,\vec{y}}$ then $i \prec_{\vec{y}} j$.
(D) If $C^*_{i,\vec{y}} = C^*_{j,\vec{y}}$ and $g_{i,\vec{y}}(\vec{y}) <_{\text{lex}} g_{j,\vec{y}}(\vec{y})$ then $i \prec_{\vec{y}} j$.

Note that conditions (C) and (D) do not determine a linear order on $M_{\vec{y}}$, since there may be distinct $i, j \in M_{\vec{y}}$ such that $C^*_{i,\vec{y}} = C^*_{j,\vec{y}}$ and $g_{i,\vec{y}}(\vec{y}) = g_{j,\vec{y}}(\vec{y})$. If this is the case, decide arbitrarily whether $i \prec_{\vec{y}} j$ or $j \prec_{\vec{y}} i$. No matter how we decide, the resulting structure $C_{t,\vec{x}}$ and mapping $g_{t,\vec{x}}$ will be the same.

Note that for every $\backsimeq$-equivalence class K, either $K \cap M_{\vec{y}} = \varnothing$ or $K \subseteq M_{\vec{y}}$. Let $N_{\vec{y}}$ be the system of representatives for the $\backsimeq$-equivalence classes in $M_{\vec{y}}$ that contains the $\preceq_{\vec{y}}$-smallest element of each class. Let i_0 be the minimal element of $N_{\vec{y}}$. We define $D^*_{\vec{y}}$ to be the structure obtained in the following three steps:

(E) For each $i \in N_{\vec{y}}$, we take a copy $C^{**}_{i,\vec{y}}$ of $C^*_{i,\vec{y}}$ and shift the universes of these copies in such a way that they are disjoint intervals of nonnegative integers arranged in the order given by $\preceq_{\vec{y}}$.
(F) We take the union of all the $C^{**}_{i,\vec{y}}$. Then for each $i \in N_{\vec{y}}$ we identify the tuple $g_{i,\vec{y}}(\vec{y})$ with the tuple $g_{i_0,\vec{y}}(\vec{y})$.
(G) We shrink the universe so that it becomes an initial segment of the positive integers.

Then $D^*_{\vec{y}}$ is an isomorphic copy of the union $D_{\vec{y}}$ of all structures $C_{i,\vec{y}}$ for $i \in N_{\vec{y}}$. Let $\vec{y}^* := g_{i_0,\vec{y}}(\vec{y})$. Observe that $D^*_{\vec{y}}$ and $\vec{y}^*$ indeed do not depend on the order $\preceq_{\vec{y}}$, as long as it satisfies (C) and (D).

Let ρ be the unique mapping from $\mathcal{Y}$ to an initial segment of the positive integers such that $\rho(\vec{y}) \leq \rho(\vec{z})$ if and only if one of the following two conditions is satisfied:

(H) $D^*_{\vec{y}} <^{\lambda}_{\text{lex}} D^*_{\vec{z}}$.
(I) $D^*_{\vec{y}} = D^*_{\vec{z}}$ and $\vec{y}^* \leq_{\text{lex}} \vec{z}^*$.

Then $\rho(\vec{y}) = \rho(\vec{z})$ if and only if $D^*_{\vec{y}} = D^*_{\vec{z}}$ and $\vec{y}^* = \vec{z}^*$. Let $r := \max\{\rho(\vec{y}) \mid \vec{y} \in \mathcal{Y}\}$.

We let $A_{t,\vec{x}}$ be the μ-expansion of $C[\beta(t)]$ defined as follows:

(J) For all $i \in [a]$ we define $P_i(A_{t,\vec{x}})$ by

$$P_i^{A_{t,\vec{x}}}(\vec{y}) := \begin{cases} 1 & \text{if } i = s_t \text{ and } \vec{y} = \vec{x}, \\ 0 & \text{otherwise,} \end{cases}$$

for all $\vec{y} \in \beta(t)^i$.
(K) For all $i \in [a]$ we define $Q_i(A_{t,\vec{x}})$ by

$$Q_i^{A_{t,\vec{x}}}(\vec{y}) := \begin{cases} \rho(\vec{y}) & \text{if } \vec{y} \in \mathcal{Y}, \\ 0 & \text{otherwise,} \end{cases}$$

for all $\vec{y} \in \beta(t)^i$.

Observe that the Gaifman graph of $A_{t,\vec{x}}$ is $\tau(t)$, because the sets $\tilde{x} = \sigma(t)$ and $\tilde{y} = \sigma(u_i)$ for all $\vec{y} \in \mathcal{Y}$ are cliques in $\tau(t)$. Hence the canonisation mapping $\mathfrak{a}$ is applicable to $A_{t,\vec{x}}$. Let $A^*_{t,\vec{x}} := \mathfrak{a}(A_{t,\vec{x}})$.

We define the structure $C^*_{t,\vec{x}}$ as follows: We take the disjoint union of $A^*_{t,\vec{x}}$ with copies of the structures $D^*_{\vec{y}}$ for $\vec{y} \in \mathcal{Y}$. These copies are chosen such that their universes are consecutive intervals of positive integers and such that $D^*_{\vec{y}}$ comes before $D^*_{\vec{z}}$ if $\rho(\vec{y}) < \rho(\vec{z})$. Let C^1 be the resulting structure. Then for each tuple $\vec{z} \in V(A^*_{t,\vec{x}})$ with $q := Q_{|\vec{z}|}^{A^*_{t,\vec{x}}}(\vec{z}) \in [1, r]$ we choose a tuple $\vec{y} \in \mathcal{Y}$ with $\rho(\vec{y}) = q$ and identify the copy of $\vec{y}^*$ in the copy of $D^*_{\vec{y}}$ in C^1 with $\vec{z}$. Of course there may be several $\vec{z}$ with $Q_{|\vec{z}|}^{A^*_{t,\vec{x}}}(\vec{z}) = q$, say, $\vec{z}_1 <_{\text{lex}} \vec{z}_2 <_{\text{lex}} \dots <_{\text{lex}} \vec{z}_\ell$. Then there are $\vec{y}_1, \dots, \vec{y}_\ell \in \mathcal{Y}$ with $\rho(\vec{y}_j) = q$. For all these, the structures $D^*_{\vec{y}_j}$ are isomorphic and their copies appear consecutively in C^1. We identify $\vec{z}_j$ with the copy of $\vec{y}^*_j$ in the copy of $D^*_{\vec{y}_j}$. After doing all these identifications, we shrink the universe of the structure so that it is an initial segment of the positive integers. Let C^2 be the resulting μ-structure, and let $C^*_{t,\vec{x}}$ be the λ-restriction of C^2. If $s_t = 0$ (and thus $\vec{x}$ is the empty tuple) we let $g_{t,\vec{x}}$ be the empty mapping. Otherwise, there is a unique tuple $\vec{x}^* = (x^*_1, \dots, x^*_{s_t}) \in P_{s_t}(A^*_{t,\vec{x}})$. We define $g_{t,\vec{x}}$ by letting $g_{t,\vec{x}}(x_i) := x^*_i$ for all all $i \in [s_t]$.

To define the canonical copy $\mathfrak{c}(C)$ of C, we let $M \subseteq V(D)$ be the set of all nodes t with $\sigma(t) = \varnothing$ and $\gamma(t) = V(C)$. Such nodes exist by (TL.5), because G_C is connected. We look at the set $\mathcal{M}$ of all structures C^*_t for nodes $t \in T$. (Here we write C^*_t instead of $C^*_{t,()}$, omitting the empty tuple enumerating $\sigma(t) = \varnothing$.) By (B), all structures in $\mathcal{M}$ are isomorphic to C. We let $\mathfrak{c}(C)$ be the $\leq^{\lambda}_{\text{lex}}$-minimal structure in $\mathcal{M}$.

It is important to note that our construction is "invariant", that is, completely determined by the structure C and the decomposition Δ_{G_C}. The only freedom we have during the construction is in the exact order of the children $u_1, \dots, u_m$ of t, but we have already noted that conditions (C) and (D) restrict the choices we can make in such a way that they do no longer matter because the resulting structures will be isomorphic. The invariance of Δ implies that if f is an isomorphism from C to a λ-structure C' then, letting

$\Delta_{G'_C} = (D', \sigma', \alpha')$, there is an isomorphism g from D to D' such that for all $t \in V(D)$ and all enumerations $\vec{x}$ of $\sigma(t)$ the restriction of f to $\gamma(t)$ is an isomorphism from $C_{t,\vec{x}}$ to $C_{g(t),f(\vec{x})}$. By the invariance of our construction, it follows that $C^*_{t,\vec{x}} = C'^*_{g(t),f(\vec{x})}$ and $g_{t,\vec{x}}(\vec{x}) = g_{g(t),f(\vec{x})}(f(\vec{x}))$. This implies that $\mathcal{M} = \mathcal{M}'$, where $\mathcal{M}'$ is defined from C' and $\Delta_{G'_C}$ in the same way as $\mathcal{M}$ is defined from C and Δ_{G_C}, and thus $\mathfrak{c}(C) = \mathfrak{c}(C')$.

As the decomposition mapping Δ is polynomial time computable, the canonization mapping $\mathfrak{c}$ is polynomial time computable as well. □

Now we are ready to prove the main algorithmic result of the paper (which proves Theorem 1.3 in the introduction):

THEOREM 9.7. *For every graph H, the class of graphs excluding H as topological subgraph admits polynomial time strong canonisation.*

PROOF. Choose the constants a, b, c, d, e as in the Invariant Decomposition Theorem 8.3. Let $\mathcal{A}_1$ be the class of all graphs G with $|V(G)| \leq b$, and let $\mathcal{A}_2$ be the class of all graphs G with $K_x \not\leq G$, and let $\mathcal{A}_3$ be the class of all graphs G such that at most c vertices of G have degree higher than d. The class $\mathcal{A}_1$ trivially admits polynomial time strong canonisation. The class $\mathcal{A}_2$ admits polynomial time strong canonisation by Lemma 9.4. The class $\mathcal{A}_3$ admits polynomial time strong canonisation by Corollary 9.5. As $\mathcal{A}_1$ and $\mathcal{A}_3$ are polynomial time decidable ($\mathcal{A}_2$ is as well, but we do not need this), it follows that the class $\mathcal{A} := \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_2$ admits polynomial time strong canonisation as well. That is, the canonization algorithm for $\mathcal{A}$ uses the algorithm for $\mathcal{A}_1$ if $|V(G)| \leq b$; otherwise it uses the algorithm for $\mathcal{A}_3$ if there are at most c vertices having degree higher than d; otherwise it uses the algorithm for $\mathcal{A}_2$.

By the Invariant Decomposition Theorem 8.3 the class of H-topological subgraph free graphs admits polynomial time invariant treelike decompositions over $\mathcal{A}$ of adhesion a. Hence by the Lifting Lemma 9.6, the class of graphs excluding H as topological subgraph admits polynomial time strong canonisation. □

10. REFERENCES

[1] O. Amini, F. V. Fomin, and S. Saurabh. Implicit branching and parameterized partial cover problems. *J. Comput. Syst. Sci.*, 77(6):1159–1171, 2011.

[2] L. Babai and E. Luks. Canonical labeling of graphs. In *STOC*, pages 171–183, 1983.

[3] H. Bodlaender. Polynomial algorithms for graph isomorphism and chromatic index on partial k-trees. *Journal of Algorithms*, 11:631–643, 1990.

[4] E. D. Demaine, M. Hajiaghayi, and K. Kawarabayashi. Contraction decomposition in H-minor-free graphs and algorithmic applications. In *STOC*, pages 441–450, 2011.

[5] E. D. Demaine, M. T. Hajiaghayi, and K. Kawarabayashi. Algorithmic graph minor theory: Decomposition, approximation, and coloring. In *FOCS*, pages 637–646, 2005.

[6] F. Dorn, F. V. Fomin, and D. M. Thilikos. Catalan structures and dynamic programming in H-minor-free graphs. In *SODA*, pages 631–640, 2008.

[7] Z. Dvorak, D. Král, and R. Thomas. Deciding first-order properties for sparse graphs. In *FOCS*, pages 133–142, 2010.

[8] I. S. Filotti and J. N. Mayer. A polynomial-time algorithm for determining the isomorphism of graphs of fixed genus. In *STOC*, pages 236–243, 1980.

[9] M. Grohe. Descriptive complexity, canonisation, and definable graph structure theory. Manuscript. `http://www2.informatik.hu-berlin.de/~grohe/cap`.

[10] M. Grohe. Local tree-width, excluded minors, and approximation algorithms. *Combinatorica*, 23(4):613–632, 2003.

[11] M. Grohe. Definable tree decompositions. In *LICS*, pages 406–417, 2008.

[12] M. Grohe, K. Kawarabayashi, D. Marx, and P. Wollan. Finding topological subgraphs is fixed-parameter tractable. In *STOC*, pages 479–488, 2011.

[13] K. Kawarabayashi, E. D. Demaine, and M. Hajiaghayi. Additive approximation algorithms for list-coloring minor-closed class of graphs. In *SODA*, pages 1166–1175, 2009.

[14] K. Kawarabayashi and B. Mohar. Approximating the list-chromatic number and the chromatic number in minor-closed and odd-minor-closed classes of graphs. In *STOC*, pages 401–416, 2006.

[15] K. Kawarabayashi and B. Mohar. Some recent progress and applications in graph minor theory. *Graphs and Combinatorics*, 23(1):1–46, 2007.

[16] K. Kawarabayashi and P. Wollan. A shorter proof of the graph minor algorithm: the unique linkage theorem. In *STOC*, pages 687–694, 2010.

[17] K. Kawarabayashi and P. Wollan. A simpler algorithm and shorter proof for the graph minor decomposition. In *STOC*, pages 451–458, 2011.

[18] E. Luks. Isomorphism of graphs of bounded valance can be tested in polynomial time. *J. Comput. Syst. Sci.*, 25:42–65, 1982.

[19] W. Mader. Homomorphieeigenschaften und mittlere Kantendichte von Graphen. *Math. Ann.*, 174:265–268, 1967.

[20] G. Miller. Isomorphism of k-contractible graphs. A generalization of bounded valence and bounded genus. *Information and Control*, 56:1–20, 1983.

[21] G. L. Miller. Isomorphism testing for graphs of bounded genus. In *STOC*, pages 225–235, 1980.

[22] I. N. Ponomarenko. The isomorphism problem for classes of graphs that are invariant with respect to contraction. *Zap. Nauchn. Sem. Leningrad. Otdel. Mat. Inst. Steklov. (LOMI)*, 174(Teor. Slozhn. Vychisl. 3):147–177, 182, 1988. In Russian.

[23] N. Robertson and P. Seymour. Graph minors X. Obstructions to tree-decomposition. *Journal of Combinatorial Theory, Series B*, 52:153–190, 1991.

[24] N. Robertson and P. Seymour. Graph minors XIII. The disjoint paths problem. *Journal of Combinatorial Theory, Series B*, 63:65–110, 1995.

[25] N. Robertson and P. Seymour. Graph minors XVI. Excluding a non-planar graph. *Journal of Combinatorial Theory, Series B*, 77:1–27, 1999.