Canonical tree-decompositions of finite graphs
II. Essential parts

J. Carmesin R. Diestel M. Hamann F. Hundertmark

October 3, 2014

Abstract

In Part I of this series we described three algorithms that construct canonical tree-decompositions of graphs which distinguish all their k-blocks and tangles of order k. We now establish lower bounds on the number of parts in these decompositions that contain such a block or tangle, and determine conditions under which such parts contain nothing but a k-block.

Introduction

A *k-block* in a graph G, where k is any positive integer, is a maximal set X of at least k vertices such that no two vertices $x, x' \in X$ can be separated in G by fewer than k vertices other than x and x'. Thus, k-blocks for large k can be thought of as highly connected pieces of a graph, but their connectivity is measured not in the subgraph they induce but in the ambient graph.

Another concept of highly connected pieces of a graph, formally quite different from k-blocks, is the notion of a tangle proposed by Robertson and Seymour [8]. Tangles are not defined directly in terms of vertices and edges, but indirectly by assigning to every low-order separation of the graph one of its two sides, the side in which 'the tangle' is assumed to sit. In order for this to make sense, the assignments of sides have to satisfy some consistency constraints, in line with our intuition that one tangle should not be able to sit in disjoint parts of the graph at once.

In a fundamental paper on graph connectivity and tree structure, Hundertmark [6] showed that high-order blocks and tangles have a common generalization, which he called 'profiles'. These also work for discrete structures other than graphs. We continue to work with profiles in this paper. All the reader needs to know about profiles is explained in Part I of this paper [1].

In Part I we described a family of algorithms which construct, for any finite graph G and $k \in \mathbb{N}$, a tree-decomposition of G that has two properties: it distinguishes all the k-blocks and tangles of order k in G, so that distinct blocks or tangles come to sit in distinct parts of the decomposition, and it is canonical in that the map assigning this decomposition to G commutes with graph isomorphisms.

In this follow-up to [1], we study these decompositions in more detail. Given k, let us call a part of such a decomposition *essential* if it contains a k-block or accommodates a tangle of order k. (Precise definitions will follow.) Since the aim of our tree-decompositions is to display how G can be cut up into its highly connected pieces, ideally every part of such a decomposition would be essential, and the essential parts containing a k-block would contain nothing else. (This makes no sense for tangles, since they cannot be captured by a set of vertices.)

Neither of these aims can always be attained. Our objective is to see when or to which extent they can. After providing in Section 1 some background on how tree-decompositions relate to oriented separation systems, we devote Section 2 to establishing upper bounds on the number of inessential parts in a canonical tree-decomposition of a graph that distinguishes all its k-profiles. These bounds depend in interesting ways on the algorithm chosen to find the decomposition. All the bounds we establish are sharp.

In Section 3 we investigate to what extent the decomposition parts containing a k-block can be required to contain nothing else. It turns out that there can be k-blocks that never occur as entire parts in a tree-decomposition of adhesion $< k$, due to a local obstruction in terms of the way in which these blocks are separated from the rest of G. However, one can show that this is the only obstruction. We find a condition by which such blocks can be identified, which leads to the following best possible result: for every k, every finite graph has a canonical tree-decomposition that distinguishes all its k-profiles efficiently and in which all those k-blocks are a part that occur as a part in *some* tree-decomposition of adhesion $< k$ [4]. Finally, we establish some sufficient global conditions on G to ensure that *every* decomposition part containing a k-block contains nothing else. (So these conditions imply that there are no local obstructions to this as found earlier, for any k-block.)

In order to read this paper with ease, the reader should be familiar with [1]; in particular with the terminology introduced in Section 2 there, the notions of a task and a strategy as defined in Section 3, and the notion of a k-strategy as defined in Section 4. The proofs in [1] need not be understood in detail, but Examples 1 and 4 make useful background.

Readers interested in k-blocks as such may refer to [2], where we relate the greatest number k such that G has a k-block to other graph invariants.

Throughout this paper, we consider a fixed finite graph $G = (V, E)$.

1 Orientations of decomposition trees

By [1, Theorem 2.2], every nested proper separation system N of our graph $G = (V, E)$ gives rise to a tree-decomposition $(T, \mathcal{V})$ that *induces* it, in that the edges of the decomposition tree T correspond to the separations in N. How exactly $(T, \mathcal{V})$ can be obtained from N is described in [3]. In this paper we shall be concerned with how profiles – in particular, blocks and tangles – correspond to nodes of T. This correspondence will be injective – distinct blocks or tangles will ‘live in’ distinct nodes of T – but it will not normally be onto: only some of the nodes of T will accommodate a block or tangle (all of some fixed order k).

Our aim in this section is to show that every profile P living in a node of T defines a consistent orientation of $E(T)$ (towards that node), or equivalently of N (namely, $N \cap P$),[1] and that the set of *all* consistent orientations of N corresponds bijectively to the nodes of T. Let $\mathcal{V} = (V_t)_{t \in T}$.

As $(T, \mathcal{V})$ induces N, there is for every separation $(A, B) \in N$ an oriented edge $e = t_A t_B$ of T such that, if T_A denotes the component of $T - e$ that contains t_A and T_B denote the component containing t_B, we have

$$(A, B) = \Big(\bigcup_{t \in T_A} V_t \, , \, \bigcup_{t \in T_B} V_t \Big).$$

If $(T, \mathcal{V})$ was obtained from N as in [1, Theorem 2.2], then e is unique, and we say that it *represents* (A, B) *in* T.[2]

Every node $t \in T$ induces an orientation of the edges of T, towards it. This corresponds as above to an orientation $O(t)$ of N,

$$O(t) := \{(A, B) \in N \mid t \in T_B\},$$

from which we can reobtain the part V_t of $(T, \mathcal{V})$ as

$$V_t \; = \bigcap_{(A,B) \in O(t)} B. \tag{1}$$

We say that t *induces* the orientation $O(t)$ of N, and that the separations in $O(t)$ are *oriented towards* t.

Distinct nodes $t, t' \in T$ induce different orientations of N, since these orientations disagree on every separation that corresponds to an edge on the path tTt'. Also clearly, not all orientations of N are induced by a node of T. But it is interesting in our context to see which are:

Theorem 1.1. (i) *The orientations of N that are induced by nodes of T are precisely the consistent orientations of N.*

(ii) *An orientation of the set of all proper $(< k)$-separations of G orients the separations induced by any tree-decomposition of adhesion $< k$ towards a node of its decomposition tree if and only if it is consistent.*

Proof. (i) Let O be an orientation of N that is not induced by a node in T, and consider the corresponding orientation of (the edges of) T. Then there are edges e, e' of T that point in opposite directions. Indeed, follow the orientated edges of T to a sink t; this exists since T is finite. As t does not induce O, some oriented edge $e' = t't''$ has t lie in the component of $T - e'$ that contains t'.

[1] Recall that profiles are consistent orientations of separation systems such as N satisfying a further axiom (P). A set of oriented tree-edges or of separations of G is *consistent* if no two of them point away from each other; see [1] for the formal definition.

[2] In general if e is not unique, we can make it unique by contracting all but one of the edges of T inducing a given partition in N, merging the parts corresponding to the nodes of contracted edges.

Then $T - e'$ contains a t'–t path. Its last edge e is oriented towards t, by the choice of t. The separations $(A, B), (C, D) \in O$ represented by e and e' then satisfy $(B, A) \le (C, D)$, so O is inconsistent.

For the converse implication suppose an orientation $O(t)$ induced by some $t \in T$ is inconsistent. Then there are $(A, B), (C, D) \in O(t)$ with $(D, C) \le (A, B)$. Let e be the oriented edge of T representing (A, B), and let f be the oriented edge representing (C, D).

Consider the subtrees T_A, T_B, T_C, T_D of T. Note that $T_B \cap T_D$ contains t, by definition of $O = O(t)$, and hence contains the component T_t of $T - e - f$ containing t.

If $f \in T_A$, then T_B is a connected subgraph of $T - f$ containing t, and hence contained in T_D. With $T_B \subseteq T_D$ we also have $B \subseteq D$. But now $(D, C) \le (A, B)$ implies $B \subseteq D \subseteq A$, and so (A, B) is not a proper separation. But it is, because $(A, B) \in N$. Hence $f \in T_B$, and similarly $e \in T_D$.

Let us show that $t_B, t_D \in T_t$. Suppose f lies on the path in T from t to t_B. Then this path traverses f from t_D to t_C, since its initial segment from t to f lies in T_D (the component of $T - f$ containing t) and hence ends in t_D. But then $e \in T_C$, contrary to what we have shown. Thus $f \notin tTt_B$, and clearly also $e \notin tTt_B$. Therefore $t_B \in T_t$, and similarly $t_D \in T_t$.

Since $f \notin T_A$, we know that T_A is a connected subgraph of $T - f$ containing an end of e. Adding e to it we obtain a connected subgraph of $T - f$ that contains both ends of e and therefore meets T_t, and adding T_t too we obtain a connected subgraph of $T - f$ that contains both T_A and t_D. Therefore $T_A \subseteq T_D$, and thus $A \subseteq D$. Analogously, $C \subseteq B$. But now $(D, C) \le (A, B)$ implies both $A \subseteq D \subseteq A$ and $C \subseteq B \subseteq C$, giving $(A, B) = (D, C)$. But then O contains both (C, D) and (D, C), which contradicts its definition as an orientation of N.

(ii) If a given orientation of the set S_k of all proper $(< k)$-separations of G is consistent, then so is the orientation it induces on N. By (i), this orientation of N orients it towards a node of the decomposition tree.

Conversely, if an orientation of S_k is inconsistent, then this is witnessed by separations $(A, B), (C, D) \in S_k$ with $(C, D) \le (A, B)$ such that (A, B) is oriented towards B but (C, D) is oriented towards C. By [1, Theorem 2.2], $N = \{(A, B), (B, A), (C, D), (D, C)\}$ is induced by a tree-decomposition $(T, \mathcal{V})$. Since the orientation $\{(D, C), (A, B)\}$ which our given orientation of S_k induces on N is inconsistent, we know from (i) that it does not orient N towards any node of T. □

Theorem 1.1 (i) implies in particular that any profile P which orients N defines a unique node $t \in T$: the t that induces its N-profile $P \cap N = O(t)$. We say that P *inhabits* this node t and the corresponding part V_t. If P is a k-block profile, induced by the k-block X, say, then this is the case if and only if $X \subseteq V_t$.

Given a set $\mathcal{P}$ of profiles, we shall call a node t of T and the corresponding part V_t *essential (wrt. $\mathcal{P}$)* if there is a profile in $\mathcal{P}$ which inhabits t.

Nodes t such that $V_t \subseteq A \cap B$ for some $(A, B) \in N$ are called *hub nodes*; the node t itself is then a *hub*. Example 2 in [1] shows that distinct hub nodes

$t, t' \in T$ may have the same hub $V_t = V'_t$. So the bijection established by Theorem 1.1 does not induce a similar correspondence between the consistent orientations of N and the parts of $(T, \mathcal{V})$ as a set, only as a family $\mathcal{V} = (V_t)_{t \in T}$. This is illustrated by [3, Figure 7].

Theorem 1.1 (ii) will not be needed in the rest of this paper. But it is interesting in its own right, in that it provides a converse to the following well-known fact in graph minor theory. Every haven [9], preference [7] or bramble [5] of order $\geq k$ in G orients the set S_k of all ($< k$)-separations of G (e.g., 'towards' that bramble). In particular, it orients the separations induced by any tree-decomposition of adhesion $< k$, and it orients these towards a node of that decomposition tree. But this fact has no converse: while it is always possible to orient S_k in such a way that the separations induced by any tree-decomposition of adhesion $< k$ are oriented towards a node t of the decomposition tree – take[3] any consistent orientation of S_k and apply Theorem 1.1 (ii) – this orientation of S_k need not be a haven or preference of order k: there may be no bramble of order $\geq k$ 'living in' t.[4]

Theorem 1.1 (ii) shows that the consistent orientations of S_k, which are generalizations of havens or preferences of order k, are the unique weakest-possible such generalization that still orients all tree-decompositions of adhesion $< k$ towards a node.

2 Bounding the number of inessential parts

Let $k \in \mathbb{N}$, and let $\mathcal{P}$ be a set of k-profiles of our graph G, both fixed throughout this section. Whenever we use the term 'essential' in this section, this will be with reference to this set $\mathcal{P}$.

Any canonical tree-decomposition distinguishing $\mathcal{P}$ has at least $|\mathcal{P}|$ essential parts, one for every profile in $\mathcal{P}$. Our aim in this section is to bound its number of inessential parts in terms of $|\mathcal{P}|$.

Variants of [1, Example 1] show that no such bounds exist if we ever use a strategy that has all, $\mathsf{all_r}$, ext or loc among its values, so we confine ourselves to strategies with values in $\{\mathsf{ext_r}, \mathsf{loc_r}\}$.

The definition of the parts of a tree-decomposition $(T, \mathcal{V})$ being somewhat complicated (see Section 1), rather than bounding the number $|\mathcal{V}| - |\mathcal{P}|$ of inessential parts of $(T, \mathcal{V})$ directly, we shall bound the number $|N|$ instead. Since $\frac{1}{2}|N|$ is the number of edges of T – as N contains 'oriented' separations, every edge of T appears twice – and $\frac{1}{2}|N| + 1$ its number of nodes, the number of inessential parts will then be $\frac{1}{2}|N| + 1 - |\mathcal{P}|$.

Our aim, then, will be to choose a strategy that minimizes $|N|$. Our strategies should therefore take values in $\{\mathsf{ext_r}, \mathsf{loc_r}\}$ only, i.e., we should reduce

[3] Every separation system has a consistent orientation; see [1].

[4] For example, identify three copies of K^5 in one vertex v, and orient every (< 2)-separation towards the side that contains two of these K^5. This is a consistent orientation of S_2 that is not a 2-haven or 2-preference and is not induced by a bramble of order ≥ 2, but which still orients the 1-separations of any tree-decomposition of adhesion 1 towards a node t (whose corresponding part could be either a K^5 or a K^1 hub).

our tasks before we tackle them, by deleting separations that do not distinguish any profiles in $\mathcal{P}$. Moreover, for a single reduced task $(S,\mathcal{P})$ we have $\mathsf{ext}(S,\mathcal{P}) \subseteq \mathsf{loc}(S,\mathcal{P})$ by [1, (11)], i.e., every separation chosen by ext is also chosen by loc. This suggests that the overall strategy Ext, which only uses $\mathsf{ext_r}$, should also return fewer separations than Loc, which only uses $\mathsf{loc_r}$ – perhaps substantially fewer, since if we select fewer separations at each step we also have more interim steps in which we reduce.

Surprisingly, this is not the case. Although our general bounds for Ext are indeed better than those for Loc (or the same, which is already a surprise), Example 2 below will show Loc yields better results than Ext for some graphs.

Let $|\mathcal{P}| =: p$. For single tasks $(S,\mathcal{P})$, we obtain the following bounds on $|N|$:

Lemma 2.1. *For every feasible task* $(S,\mathcal{P})$ *we have*

$$2(p-1) \le |N_{\mathsf{Ext}}(S,\mathcal{P})| \le 2p, \text{ and} \tag{2}$$

$$2(p-1) \le |N_{\mathsf{Loc}}(S,\mathcal{P})| \le 4(p-1). \tag{3}$$

Proof. The two lower bounds, which in fact hold for any strategy, follow from the fact that N gives rise to a tree-decomposition $(T,\mathcal{V})$ that induces it and distinguishes $\mathcal{P}$: this means that $|N| = 2\,(|T|-1) \ge 2(p-1)$.

Let us now prove the upper bound in (2), by induction on p. Let $(R,\mathcal{P})$ be the reduction of $(S,\mathcal{P})$. If $p \le 1$ then $R = \emptyset$, so the statement is trivial. Now assume that $p \ge 2$. Then $S \ne \emptyset$, since S distinguishes $\mathcal{P}$. Let $\mathcal{P}^{\mathrm{e}}$ be the set of profiles in $\mathcal{P}$ that are extremal in $(R,\mathcal{P})$. By [1, Lemma 3.1] we have $\mathcal{P}^{\mathrm{e}} \ne \emptyset$. Then

$$N_{\mathsf{Ext}}(S,\mathcal{P}) = \ N \cup \bigcup_{O \in \mathcal{O}_N} N_{\mathsf{Ext}}(S_O,\mathcal{P}_O)\,, \tag{4}$$

by definition of $N_{\mathsf{Ext}}(S,\mathcal{P})$, where $N = \mathsf{ext_r}(S,\mathcal{P})$. Every extremal $P \in \mathcal{P}$ is distinguished from all the other profiles in $\mathcal{P}$ by the separation (A,B) for which $P = P_{(A,B)}$, so P lies in a singleton class $\mathcal{P}_O = \{P\}$. Then $(S_O,\mathcal{P}_O)$ reduces to $(\emptyset,\mathcal{P}_O)$, giving $N_{\mathsf{Ext}}(S_O,\mathcal{P}_O) = \emptyset$ for these $O \in \mathcal{O}_N$. By the uniqueness of $P_{(A,B)}$ in [1, Lemma 3.2], no separation in N distinguishes two non-extremal profiles from $\mathcal{P}$. So there is at most one other partition class $\mathcal{P}_O$ with $O \in \mathcal{O}_N$. If such a $\mathcal{P}_O$ exists it satisfies $\mathcal{P}_O = \mathcal{P} \setminus \mathcal{P}^{\mathrm{e}}$, and if it is non-empty the $O \in \mathcal{O}_N$ giving rise to it is unique. Therefore

$$N_{\mathsf{Ext}}(S,\mathcal{P}) = \mathsf{ext_r}(S,\mathcal{P}) \cup N_{\mathsf{Ext}}(S_O,\mathcal{P} \setminus \mathcal{P}^{\mathrm{e}})$$

for this O if $\mathcal{P} \setminus \mathcal{P}^{\mathrm{e}} \ne \emptyset$, and $N_{\mathsf{Ext}}(S,\mathcal{P}) = \mathsf{ext_r}(S,\mathcal{P})$ otherwise. In the first case we have

$$|N_{\mathsf{Ext}}(S_O,\mathcal{P} \setminus \mathcal{P}^{\mathrm{e}})| \le \ 2\,|\mathcal{P} \setminus \mathcal{P}^{\mathrm{e}}|$$

by the induction hypothesis, and in both cases we have $|\mathsf{ext_r}(S,\mathcal{P})| \le 2\,|\mathcal{P}^{\mathrm{e}}|$ by [1, Lemma 3.2 and (9)]. This completes the proof of (2).

For a proof of the upper bound in (3) let $(T,\mathcal{V})$ be a tree-decomposition of G that induces $N_{\mathsf{Loc}}(S,\mathcal{P})$ as in [1, Theorem 2.2]. Since $N_{\mathsf{Loc}}(S,\mathcal{P})$ contains only

$\mathcal{P}$-relevant separations, all the leaves of T are essential. Furthermore, we shall prove the following:

$$\textit{For every edge } e = t_1t_2 \textit{ of } T, \textit{ either } t_1 \textit{ or } t_2 \textit{ is essential.} \tag{5}$$

Before we prove (5), let us show how it helps us establish the upper bound in (3). If (5) holds, then all the neighbours of an inessential node are essential. Let T' be obtained from T by deleting each inessential node and adding an edge from one of its neigbours to all its other neighbours. Let us show that

$$T' \textit{ has } p \textit{ nodes and at least half as many edges as } T. \tag{6}$$

The first of these assertions holds by definition of T and p. For the second, note that for each inessential node we delete we lose exactly one edge. So to prove the second claim in (6) it suffices to show that T has at most $\|T\|/2$ inessential nodes. But this follows from (5) and the fact that the leaves of T are essential: every inessential node has at least two incident edges, and no edge is counted twice in this way (i.e., is incident with more than one inessential node).

By (6), T has at most $2(p-1)$ edges. Since $N_{\mathsf{Loc}}(S, \mathcal{P})$ is induced by $(T, \mathcal{V})$, this will establish the upper bound in (3).

So let us prove (5). Suppose T has an edge $e = t_1t_2$ with neither t_i essential. Let $(A, B) \in N_{\mathsf{Loc}}(S, \mathcal{P})$ be the separation which e induces. Let T_A denote the component of $T-e$ that contains t_1, and let T_B be the component containing t_2.

At the time (A, B) was chosen by Loc we had a nested proper separation system N and a consistent orientation O of N such that $(A, B) \in \mathsf{loc}_{\mathsf{r}}(S_O, \mathcal{P}_O)$. (When $N = \emptyset$ at the start, we have $\mathsf{loc}_{\mathsf{r}}(S_O, \mathcal{P}_O) = (S, \mathcal{P})$.) So there is a profile $P \in \mathcal{P}_O$ such that (A, B) or (B, A) is maximal in $(P \cap S_O, \leq)$, say (A, B). By the definition of a task, P orients S. By Lemma 1.1, therefore, P inhabits a unique node $t \in T$, making it essential. Then $(A, B) \in O(t)$, and hence $t \in T_B$. Since t_2 inessential by assumption, $t \neq t_2$.

The last edge e' on the t_2–t path in T induces a separation $(C, D) \in O(t) \subseteq P$, and $(A, B) \leq (C, D)$, or equivalently, $(D, C) \leq (B, A)$. Since (A, B) is $\mathcal{P}_O$-relevant there exists $P' \in \mathcal{P}_O$ with $(B, A) \in P'$. Then $(D, C) \in P'$, since P' orients all of S consistently. But then (C, D) splits O, and thus lies in S_O. This contradicts the maximality of (A, B), completing the proof of (5) and hence of (3). □

It is easy to see that the upper bounds in Lemma 2.1 are tight. For example, if G consists of n disjoint large complete graphs threaded on a long path, then for $k = 3$ the canonical tree-decomposition produced by Loc will have n essential parts consisting of these complete graphs and $n-1$ inessential parts consisting of the paths between them. When n is even, this example also shows that the upper bound for Ext is best possible. In fact, the following example shows that the upper bound in Lemma 2.1 (i) is best possible for all canonical tree-decompositions (regardless of which strategy is used to produce it):

Example 1. Let G consist of an n-cycle C together with n large complete graphs $K_1, \dots, K_n$ each intersecting C in one edge and otherwise disjoint. Then, for $n \geq 3$ and $k = 3$, any canonical tree-decomposition of G of adhesion $< k$ either has exactly one part or exactly the parts $C, K_1, \dots, K_n$. This is because the 2-separations of G that induce 2-separations of C cannot be induced by a canonical tree-decomposition of G, since they cross their translates under suitable automorphisms of G.

It is more remarkable, perhaps, that the upper bound in Lemma 2.1 (i) is so low: that at most one part of the tree-decomposition is not inhabited by a profile from $\mathcal{P}$. For if G is $(k-1)$-connected and S is the set of all proper $(< k)$-separations, then every task $(S, \mathcal{P})$ is feasible [1, Lemma 4.1], and hence Lemma 2.1 gives the right overall bounds.

If G is not $(k-1)$-connected, the original task $(S, \mathcal{P})$ need not be feasible, and we have to use iterated strategies. Let Ext^k denote the k-strategy all whose entries are Ext, and let Loc^k denote the k-strategy which only uses Loc. Interestingly, having to iterate costs us a factor of 2 in the case of Ext, but it does not affect the upper bound for Loc. Hence for iterated strategies the two bounds coincide:

Theorem 2.2. *Let $\mathcal{P}$ be any set of k-profiles of G, and $p := |\mathcal{P}|$. Let $N_{\mathsf{Ext}^k}(\mathcal{P})$ and $N_{\mathsf{Loc}^k}(\mathcal{P})$ be obtained with respect to the set S of all proper $(< k)$-separations.*

(i) $2(p-1) \leq |N_{\mathsf{Ext}^k}(\mathcal{P})| \leq 4\,(p-1)$

(ii) $2(p-1) \leq |N_{\mathsf{Loc}^k}(\mathcal{P})| \leq 4\,(p-1)$

(iii) *If G is $(k-1)$-connected, then $|N_{\mathsf{Ext}^k}(\mathcal{P})| \leq 2p$.*

Proof. The lower bounds for N follow as in the proof of Lemma 2.1. Statement (iii) reduces to Lemma 2.1 (i), since $\mathsf{Ext}^k = \mathsf{Ext}$ now and the entire task $(S, \mathcal{P})$ is feasible [1, Lemma 4.1].

For the proof of the upper bounds in (i) and (ii), let us define a rooted tree (T, r) that represents the recursive definition of N_{Ext^k} and N_{Loc^k}, as follows. Let

$$V(T) := \{\emptyset\} \cup \bigcup_{1 \leq \ell \leq k} \mathcal{P}_\ell\,;$$

recall that $\mathcal{P}_\ell$ for $\ell \leq k$ is the set of all ℓ-profiles of G that extend to a k-profile in $\mathcal{P}$. We select $r = \emptyset$ as the root, and make it adjacent to every $P \in \mathcal{P}_1$. For $2 \leq \ell \leq k$ we join $P \in \mathcal{P}_\ell$ to the unique $P' \in \mathcal{P}_{\ell-1}$ which it induces (i.e., for which $P' \subseteq P$). This is clearly a tree, with levels $\{\emptyset\}, \mathcal{P}_1, \dots, \mathcal{P}_k$. Let us call the vertices of T that are not in $\mathcal{P}_k$ its *internal* vertices.

The internal vertices of T correspond bijectively to the tasks which our iterated algorithm, either Ext^k or Loc^k, has to solve. Indeed, at the start the algorithm has to solve the task $(S', \mathcal{P}')$ with S' the set of proper 0-separations of G and $\mathcal{P}' = \mathcal{P}_1$ the set of 1-profiles that extend to a k-profile in $\mathcal{P}$. This task corresponds to r in that $\mathcal{P}'$ is the set of children of r. Later, for $\ell = 2, \dots, k$ recursively, the algorithm at step ℓ receives as input some tasks $(S', \mathcal{P}')$, one for every $P \in \mathcal{P}_{\ell-1}$, in which $\mathcal{P}'$ is the set of ℓ-profiles in $\mathcal{P}_\ell$ extending P, and S' is the set of proper $(\ell-1)$-separations of G that are nested with the set $N_{\ell-1}$ of nested ($< \ell - 1$)-separations distinguishing $\mathcal{P}_{\ell-1}$ which the algorithm has found so far. This task corresponds to $P \in V(T)$ in the same way, in that $\mathcal{P}'$ is the set of children of P.

Let $c(v)$ denote the number of children of an internal vertex v. Since Ext and Loc reduce every task before they solve it, the task $(S', \mathcal{P}')$ corresponding to a vertex v will add a separation to N only if $c(v) = |\mathcal{P}'| \geq 2$. Let (T', r') be obtained from (T, r) by suppressing any vertices with exactly one child; if r is suppressed, its first descendant with more than one child becomes the new root r'. The internal vertices of T' thus have degree at least 3, except that r' has degree at least 2. Let i denote the number of internal vertices of T'. Since the number of (non-root) leaves of T' is exactly p, we have at most $(p-1)$ internal vertices, that is, $i \leq p - 1$.

Now consider the construction of $N_{\mathsf{Ext}^k}(\mathcal{P})$. By (2) in Lemma 2.1, each internal vertex v of T' contributes at most $2c(v)$ separations. So there are at most twice as many separations in $N_{\mathsf{Ext}^k}(\mathcal{P})$ as there are edges in T':

$$|N_{\mathsf{Ext}^k}(\mathcal{P})| \leq 2\,\|T'\| = 2(p + i - 1) \leq 4(p-1).$$

During the construction of $N_{\mathsf{Loc}^k}(\mathcal{P})$, each internal vertex v of T' contributes at most $4(c(v) - 1)$ separations, by (3) in Lemma 2.1. Writing I for the set of internal vertices of T', we thus obtain

$$|N_{\mathsf{Loc}^k}(\mathcal{P})| \leq 4 \sum_{v \in I} \big(c(v) - 1\big) = 4\big(\|T'\| - i\big) = 4\big(|T'| - i - 1\big) = 4(p-1). \quad \square$$

It is easy to construct examples showing that all these bounds are sharp. Instead, let us give an example where Loc yields the best possible result of $2(p-1)$, while Ext does not:

Example 2. Consider the 3-connected graph with four 4-blocks shown in Figure 1. The grey bars indicate separators of chosen separations. Algorithm Ext chooses all these separations: first the two (inverse pairs of) outer separations, then the two pairs of inner separations. On the other hand, Loc will choose the three pairs of ‘straight’ separations at the first step, and no further separations thereafter. Therefore Ext chooses one pair of separations more than Loc does.

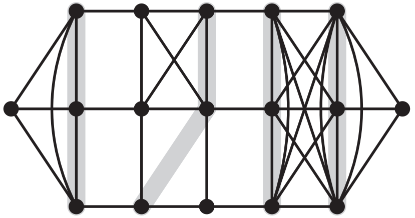

Figure 1: A graph where Loc chooses fewer separations than Ext.

3 Bounding the size of the parts

One of the first questions one may ask about canonical tree-decompositions is whether they can be chosen so as to witness the tree-width of the graph. Choosing the cycle C in Example 1 long, however, shows that this will not in general be the case: restricting ourselves to a set of separations that is invariant under all the automorphisms of G can result in arbitrarily large parts, and these need not even be essential.

However, if we restrict our attention from arbitrary k-profiles to (those induced by) k-blocks, we can try to make the essential parts small by reducing the junk they contain, the vertices contained in an essential part that do not belong to the k-block that made this part essential. Note that this aim conflicts with our earlier aim to reduce the number of inessential parts: since this junk is part of G, expunging it from the essential parts will mean that we have to have other parts to accommodate it.

In general, we shall not be able to reduce the junk in essential parts to zero unless we restrict the class of graphs under consideration. Our next example shows some graphs for which any tree-decomposition of adhesion at most k, canonical or not, has essential parts containing junk. The amount of junk in a part cannot even be bounded in terms the size of the k-block inhabiting it.

Example 3. Consider the 4-connected graph obtained by joining two adjacent vertices x, y to a K^5 as in Figure 2. This graph has a single 5-block K, the vertex set of the K^5. In any tree-decomposition of adhesion at most 4, the part containing K will contain x or y as well: since the 4-separations that separate x and y from K cross, at most one of them will be induced by the decomposition.

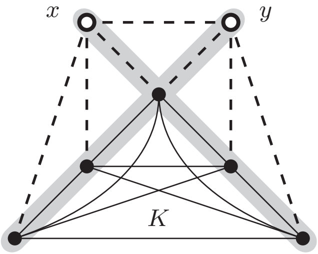

Figure 2: A K^5 with unavoidable junk attached

To increase the amount of junk in the part containing K, we can attach arbitrarily many pairs of adjacent vertices to the K^5 in the same way as we added x and y. This will not increase the size of the 5-block K, but the part containing K will also contain at least one vertex from each of those pairs.

The following theorem shows that the obstruction to obtaining essential parts without junk illustrated by the above example is, in a sense, the only such obstruction. Let us call a k-block X *well separated* in a separation system S of proper $(< k)$-separations if the k-profile $P_k(X) \cap S$ it induces in S is well separated, that is, if the maximal elements of $P_k(X) \cap S$ are nested with each other. (This fails in the example.) Recall that a separation (A, B) is *tight* if every vertex in $A \cap B$ has a neighbour in $A \setminus B$ and a neighbour in $B \setminus A$.

Theorem 3.1. *Let $1 \leq k \in \mathbb{N}$, and let S be a set of proper $(< k)$-separations that includes all the tight $(< k)$-separations. Then every graph G has a canonical*[5] *tree-decomposition all whose separations induced by tree-edges are in S such that*

(i) *distinct k-blocks lie in different parts;*

(ii) *parts containing a k-block that is well separated in S coincide with that k-block;*

(iii) *if the task $(S, \mathcal{P})$ with $\mathcal{P}$ the set of all k-block profiles is reduced and feasible, then every leaf part is a k-block.*[6]

Every such decomposition that satisfies (i), *but not necessarily* (ii) *or* (iii), *can be refined to such a tree-decomposition that also satisfies* (ii) *and* (iii).

Proof. Let $\mathcal{P}$ be the set of all k-block profiles in G. Let $N \subseteq S$ be any nested separation system that distinguishes all its k-blocks and is canonical, i.e., invariant under the automorphisms of G. Such a set N exists by [1, Theorem 4.4]. (The separations provided by that theorem are tight, and hence in S, because they are $\mathcal{P}$-essential, i.e., distinguish two profiles in $\mathcal{P}$ 'efficiently'; see [1].) Then the tree-decomposition $(T, \mathcal{V})$ that induces N by [1, Theorem 2.2] satisfies (i).

For (ii) we refine N by adding the locally maximal separations of $(S, \mathcal{P})$ and their inverses. These are nested with S by [1, Corollary 3.5]. Hence the refined separation system N' is again nested, and therefore induced by a tree-decomposition $(T', \mathcal{V}')$. This decomposition is again canonical, since the set of locally maximal separations is invariant under the automorphisms of G. Clearly, $(T', \mathcal{V}')$ still satisfies (i).

To show that $(T', \mathcal{V}')$ satisfies (ii), suppose it has a part that contains a well separated k-block X and a vertex v outside X. By the maximality of X as a $(< k)$-inseparable set, there is a separation $(A, B) \in S$ with $X \subseteq B$ and $v \in A \setminus B$. Clearly, $(A, B) \in P_k(X)$; choose (A, B) maximal in $P_k(X)$, the k-profile that X induces. Then $(A, B) \in N'$, by definition of N' and our assumption that X is well separated.[7] This contradicts our assumption that v lies in the same part of $(T', \mathcal{V}')$ as X.

To show that the decomposition $(T', \mathcal{V}')$ obtained for (ii) also satisfies (iii), consider a leaf part V_t. By the assumption in (iii), the separation $(A, B) \in N'$ that corresponds to the edge of T' at t and satisfies $B = V_t$ distinguishes two k-blocks. Let X be the k-block in V_t; it is unique, since N' distinguishes $\mathcal{P}$ but no separation in N' separates V_t. Let $P = P_k(X)$ be the k-profile that X induces. Let $(A', B') \geq (A, B)$ be maximal in S. By assumption in (iii), $B' \subseteq B$ too contains a k-block, which can only be X. Hence $(A', B') \in P$.

[5] Here, this means that the tree-decomposition will be invariant under those automorphisms of G that act on S. For example, this is the case for all the automorphisms of G if S consists of all the tight $(< k)$-separations.

[6] Recall that $(S, \mathcal{P})$ is feasible, for example, if G is $(k-1)$-connected.

[7] Compare the definition of 'locally maximal' implicit in the definition of N'.

Since $(S, \mathcal{P})$ is reduced and feasible, by assumption in (iii), [1, Lemma 3.1] implies that (A', B') is extremal in S. Hence P is extremal, and therefore well separated. We thus have $V_t = X$ by (ii). □

The idea behind allowing some flexibility for S in Theorem 3.1 is that this can make (ii) stronger by making more k-blocks well separated. For example, consider a tree-decomposition whose parts are all complete graphs K^5 and whose separations induced by tree edges all have order 3. The k-blocks for $k = 5$ are the K^5s, but none of these is well separated in the set S of all proper $(< k)$-separations, since the natural 3-separations can be extended in many ways to pairwise crossing 4-separations that will be the locally maximal separations. However all the k-blocks are well separated in the smaller set S' of all proper (< 4)-separations, which are precisely the tight $(< k)$-separations. So applying the theorem with this S' would exhibit that the essential parts of our decomposition are in fact k-blocks, a fact the theorem applied with S cannot see.

However, even with S the set of tight $(< k)$-separations, Theorem 3.1 (ii) can miss some parts in canonical tree-decompositions that are in fact k-blocks, because they are not well separated even in this restricted S:

Example 4. Let G consist of a large complete graph K to which three further large complete graphs are attached: K_1 and K_2 by separators S_1 and S_2, respectively, and K_{12} by the separator $S_1 \cap S_2$. If $|S_1| = |S_2| = k - 1$ and $S_1 \neq S_2$, the separations $(K_1 \cup K_{12}, K \cup K_2)$ and $(K_2 \cup K_{12}, K \cup K_1)$ are maximal in $P_k(K) \cap S$ for the k-block K and the set S of all tight $(< k)$-separations. They cross, since both have K_{12} on their 'small' side (Fig. 3).

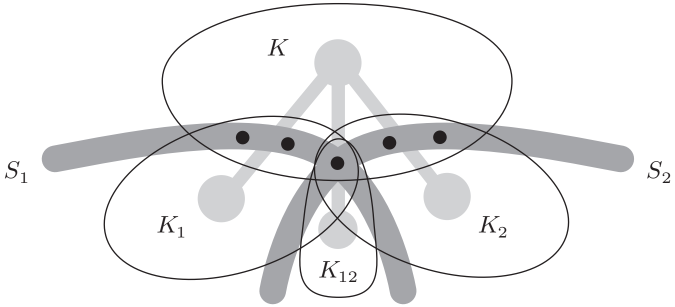

Figure 3: The 4-block K is a decomposition part but is not well separated

So K is not well separated. But the (unique) canonical tree-decomposition of G that distinguishes its k-blocks still has K as a part: its parts are the four large complete graphs, the decomposition tree being a star with centre K.

We wonder whether the notion of being well separated can be weakened, or applied to a suitable set S of $(< k)$-separations, so as to give Theorem 3.1 (ii) a converse: so that every graph has a canonical tree-decomposition that distinguishes its k-blocks, whose separations induced by decomposition tree edges are in S, and in which every well separated k-block is a part, while conversely every k-block that occurs as a part in such a tree-decomposition is well separated in S.

Here is an attempt. Given a k-block X, let $S(X)$ denote the set of all tight separations (A, B) such that $X \subseteq B$ and $A \setminus B$ is a component of $G - X$. The separations in $S(X)$ are clearly nested.[8] Write S_k for the set of all $(< k)$-separations of G. Then the condition that

$$S(X) \subseteq S_k \tag{7}$$

is a weakening of X being well separated in S_k. Indeed, if $(A, B) \in S(X)$ is not in S_k, i.e. has order $\geq k$, we can find two crossing separations both maximal in $P_k(X)$, as follows. Pick a vertex $a \in A \setminus B$. By the maximality of X as a $(< k)$-inseparable set, our vertex a can be separated from X by some $(C, D) \in S_k$, say with $a \in C \setminus D$ and $X \subseteq D$. Then $(C, D) \in P_k(X)$. Replace (C, D) with any maximal separation in $P_k(X)$ that is greater than it, and rename that separation as (C, D). Then still $a \in C \setminus D$ and $X \subseteq D$. As $A \setminus B$ is connected, and $A \cap B \subseteq X$ has size $\geq k$ although (A, B) is tight, it follows that $C \cap D$ contains a vertex $a' \in A \setminus B$. Like a, the vertex a' is separated from X by some maximal separation $(C', D') \in P_k(X)$. The separations (C, D) and (C', D') are easily seen to cross, so X is not well separated.

On the other hand, condition (7) holds for every k-block X that does occur as a part in a tree-decomposition of adhesion $< k$. Thus if (7) is still strong enough to imply that X is a part in some, or any, canonical such tree-decomposition, we shall have our desired converse of Theorem 3.1 (ii) with (7) replacing 'well separated'.

Given k, call a tree-decomposition of a graph *good* if it is canonical, has adhesion $< k$, and distinguishes all the k-blocks of G *efficiently*: any two of them are distinguished by an adhesion set whose order is minimum among all the separators in G between those blocks. The following result, which had been conjectured in the original preprint of this paper, was announced by Carmesin and Gollin [4]:

Theorem 3.2. *For every k, every finite graph has a good tree-decomposition in which every k-block X that satisfies* (7) *is a part.*

The proof of Theorem 3.2 is quite involved and builds on [3, Theorem 5.2]. However, the theorem has a corollary that can be stated with a minimum of technical overheads and emphasises the way in which it is best possible:

Corollary 3.3. *For every k, every finite graph has a good tree-decomposition that includes among its parts all k-blocks that are a part in some tree-decomposition of adhesion $< k$.*

Proof. Consider the tree-decomposition $(T, \mathcal{V})$ provided by Theorem 3.2. Let X be any k-block that occurs as a part in some good tree-decomposition. As noted earlier, this implies that X satisfies (7). By the choice of $(T, \mathcal{V})$, this means that X is also a part of $(T, \mathcal{V})$. □

[8] However, Example 4 with $X = K_1$ and $X' = K_2$ shows that for distinct k-blocks X, X' the sets $S(X)$ and $S(X')$ need not be nested: the separation $(A, B) \in S(X)$ with $A = K \cup K_2$ and $B = K_1 \cup K_{12}$ crosses the separation $(A', B') \in S(X')$ with $A' = K \cup K_1$ and $B = K_2 \cup K_{12}$.

Are there any natural conditions ensuring that *every* essential part is a k-block? (In particular, such conditions will have to rule out Example 3.) We do not know the answer to this question. But we can offer the following:

Theorem 3.4. *Assume that G is $(k-1)$-connected, and that every pair x, y of adjacent vertices has one of the following properties:*

(i) *x and y have at least $k-3$ common neighbours;*

(ii) *x and y are joined by at least $\lfloor \frac{3}{2}(k-2) \rfloor$ independent paths other than xy;*

(iii) *x and y lie in a common k-block.*

Then G has a canonical tree-decomposition of adhesion $< k$ such that every part containing a k-block is a k-block. In particular distinct k-blocks are contained in different parts.

Proof. By Theorem 3.1 it suffices to show that every element P of the set $\mathcal{P}$ of k-block profiles is well separated in the set S of all the proper $(< k)$-separations.

We do this by applying [1, Lemma 3.4]. Given $P \in \mathcal{P}$, let $(A, B), (C, D)$ be crossing separations in $P \cap S$. If the separation $(A \cup C, B \cap D)$ has order $\leq k-1$, it is in $P \cap S$ by (P), and we are done. If not, then the separation $(B \cup D, A \cap C)$ has order $< (k-1)$. Since G is $(k-1)$-connected, $(B \cup D, A \cap C)$ must be improper. This means that $A \cap C \subseteq B \cup D$, because $B \not\subseteq A$ as B contains a k-block. But since (A, B) and (C, D) cross, we cannot have $A \cap C \subseteq B \cap D$. By symmetry we may assume that there is a vertex $x \in (C \cap D) \setminus B$. As G is $(k-1)$-connected, (C, D) is tight, so x has a neighbour $y \in (A \cap B) \setminus D$. Let $e := xy$.

Suppose first that e satisfies (i). Since all common neighbours of x and y lie in $A \cap C$, this implies $k-1 \leq |A \cap C| \leq k-2$, a contradiction.

Now suppose that e satisfies (ii), and let $\mathcal{W}$ be a set of at least $\lfloor \frac{3}{2}(k-2) \rfloor$ independent x–y paths other than the edge xy. Let

$$X := (A \cap C) \setminus \{x, y\} \qquad Y := (A \cap B) \setminus C \qquad Z := (C \cap D) \setminus A\,.$$

Since $A \cap C \subseteq B \cup D$, we have

$$|X| + |Y| + |Z| \leq |A \cap B| - 1 + |C \cap D| - 1 = 2(k-2). \tag{8}$$

Every path in $\mathcal{W}$ that avoids X meets both Y and Z. As $|X| \leq (k-2) - 2$, this yields

$$|\mathcal{W}| \leq |X| + \tfrac{1}{2}\big(|Y| + |Z|\big) \leq |X| + (k-2) - \tfrac{1}{2}|X| \leq \tfrac{3}{2}(k-2) - 1,$$

a contradiction.

Finally assume that e satisfies (iii). Let X be a k-block containing x and y. As $x \notin B$ and $y \notin D$ we have $X \subseteq A \cap C$, contradicting $|A \cap C| \leq k-2$. □

For $k = 2$, Theorem 3.4 (i) implies Tutte's theorem that every 2-connected graph has a tree-decomposition whose essential parts are precisely its 3-blocks. The decomposition obtained by any strategy starting with all is the decomposition provided by Tutte [10], in which the inessential parts have cycle torsos.

References

[1] J. Carmesin, R. Diestel, M. Hamann, and F. Hundertmark. Canonical tree-decompositions of finite graphs I. Existence and algorithms. *J. Combin. Theory Ser. B*, to appear.

[2] J. Carmesin, R. Diestel, M. Hamann, and F. Hundertmark. k-Blocks: a connectivity invariant for graphs. *SIAM J. Discrete Math.*, to appear.

[3] J. Carmesin, R. Diestel, F. Hundertmark, and M. Stein. Connectivity and tree structure in finite graphs. *Combinatorica*, 34(1):1–35, 2014.

[4] Johannes Carmesin and Pascal Gollin. Canonical tree-decompositions isolating all their k-blocks. Presentation at *Hamburg workshop on graphs and matroids*, Spiekeroog 2014.

[5] R. Diestel. *Graph Theory.* Springer, 4th edition, 2010.

[6] F. Hundertmark. Profiles. An algebraic approach to combinatorial connectivity. arXiv:1110.6207, 2011.

[7] B.A. Reed. Tree width and tangles: a new connectivity measure and some applications. In R.A. Bailey, editor, *Surveys in Combinatorics.* Cambridge Univ. Press, 1997.

[8] N. Robertson and P.D. Seymour. Graph minors. X. Obstructions to tree-decomposition. *J. Combin. Theory (Series B)*, 52:153–190, 1991.

[9] P. Seymour and R. Thomas. Graph searching and a min-max theorem for tree-width. *J. Combin. Theory (Series B)*, 58(1):22–33, 1993.

[10] W. T. Tutte. *Graph Theory.* Addison-Wesley, 1984.