# Connected tree-width

Malte Müller

December 3, 2012

**Abstract**

The connected tree-width of a graph is the minimum width of a tree-decomposition whose parts induce connected subgraphs. Long cycles are examples of graphs of small tree-width but large connected tree-width.

We show that finite graphs have small connected tree-width if and only if they have small tree-width and contain no long geodesic cycle.

We further prove a qualitative duality theorem for connected tree-width: a finite graph has small connected tree-width if and only if it has no bramble whose connected covers are all large.

## 1 Introduction

Let us call a tree-decomposition $(T, (V_t)_{t \in T})$ of a graph $G$ *connected* if its parts $V_t$ are connected in $G$. For example, the standard minimum width tree-decomposition of a tree or a grid has connected parts. The *connected tree-width* $ctw(G)$ of $G$ is the minimum width that a connected tree-decomposition of $G$ can have.

Obviously $tw(G) \leq ctw(G)$, because every connected tree-decomposition is a tree-decomposition. So having large tree-width is a reason for a graph to have large connected tree-width. But it is not the only possible reason.

The parts of a (nontrivial) connected tree-decomposition of a cycle are paths. The intersection of two adjacent parts of a tree-decomposition always separates the graph, so the connected tree-decomposition of a cycle that has minimum width consists of only two parts each covering just over half of the cycle. If this cycle is contained geodesically in a larger graph $G$, its large connected tree-width will be a lower bound for the connected tree-width of $G$: there will be no shortcut that could allow us to break up the cycle into smaller connected parts. The following main theorem of this paper shows that large tree-width and large geodesic cycles are the only two reasons for a graph to have large connected tree-width.

**Theorem 1.1.** *The connected tree-width of a graph $G$ is bounded above by a function of its tree-width and the maximum length $k$ of its geodesic cycles. Specifically*

$$ctw(G) \leq tw(G) + \binom{tw(G)+1}{2} \cdot (k \cdot tw(G) - 1).$$

(If $G$ is a forest, we define $k$ to be 1)

Theorem 1.1 is qualitatively best possible in that the two reasons are independent: a large cycle (as a graph) contains a large geodesic cycle but has small tree-width, while a large grid has large tree-width but all its geodesic cycles are small.

Among the many obstructions to small tree-width there is only one that gives a tight duality theorem: the existence of a large-order bramble. A *bramble* is a set of pairwise touching connected subsets of $V(G)$, where two such subsets *touch* if they have a vertex in common or $G$ contains an edge between them. A subset of $V(G)$ *covers* (or is a *cover* of) a bramble $\mathcal{B}$ if it meets every element of $\mathcal{B}$. The *order* of a bramble is the least number of vertices needed to cover it.

**Tree-width duality theorem** (Seymour and Thomas [3])**.** *Let $k \geq 0$ be an integer. A graph has tree-width $\geq k$ if an only if it contains a bramble of order $> k$.*

Let the *connected order* of a bramble $\mathcal{B}$ be the least order of a *connected cover*, a cover of $\mathcal{B}$ spanning a connected subgraph. Since every bramble is covered by a part in any given tree-decomposition, graphs of connected tree-width $< k$ cannot have brambles of connected order $> k$. I conjecture that the converse of this holds too:

**Conjecture 1.2** (connected tree-width duality conjecture)**.** *Let $k \geq 0$ be an integer. A graph has connected tree-width $\geq k$ if and only if it contains a bramble of connected order $> k$.*

As our second main result we shall prove a qualitative version of this conjecture:

**Theorem 1.3.** *Let $k \geq 0$ be an integer. There is a function $g : \mathbb{N} \to \mathbb{N}$ such that any graph with no bramble of connected order $> k$ has connected tree-width $< g(k)$.*

The proof of Theorem 1.1 goes roughly as follows. We start with a tree-decomposition of minimum width and enlarge its parts by replacing them with connected supersets. In order to retain a tree-decomposition, we shall have to make sure that vertices which are used to make one part connected also appear in certain other parts of the tree-decomposition (compare axiom (T3) in the definition of a tree-decomposition, e.g. in [1]). Our task will be to find extensions whose sizes are bounded by a function in the maximum length of a geodesic cycle in the graph and its tree-width, regardless of its number of vertices.

All the graphs we consider in this paper will be finite and nonempty. The notation and terminology we use are explained in [1], in particular we shall assume familiarity with the basic theory of tree-decompositions as described in [1, Ch.12.3.].

The layout of this paper is as follows. In Section 2 we introduce our main technical tool for finding paths in a graph that can be used to make disconnected parts of its tree-decompositions connected: a *navigational path system*, or *navi* for short. In Section 3 we introduce tree-decompositions whose parts cannot be split, we call such tree-decompositions *atomic*. For such atomic tree-decompositions we then find cycles in the graph that are separated by its adhesion sets. In Section 4 we use those cycles to get an upper bound for the part sizes of our connected tree-decomposition which completes the proof of Theorem 1.1. In Section 5, this result will be used to prove Theorem 1.3.

## 2 Navis

How do we get an upper bound for the connected tree-width of a graph $G$? The easiest algorithmic way is to start with a tree-decomposition of minimum width and enlarge a part (which does not yet induce a connected subgraph) by adding a path of $G$ (reducing the number of components). This might result in a violation of (T3), which can be repaired by adding the corresponding vertices also to other parts. Now we can go on and make the next part (a little bit more) connected until we have a connected tree-decomposition. If we don't choose the connecting paths carefully, we might add an unbounded number of vertices to one part while repairing (T3). Take the graph and tree-decomposition indicated in Figure 1, for instance. If we choose the path containing $x_i$ for making $V_{t_i}$ connected (for every $i$), we will have to add all the $x_i$ to $V_{t_0}$ while repairing (T3), because $V_{t_0}$ lies between the part containing $x_i$ and $V_{t_i}$ which contains $x_i$ as well (after we added the connecting path).

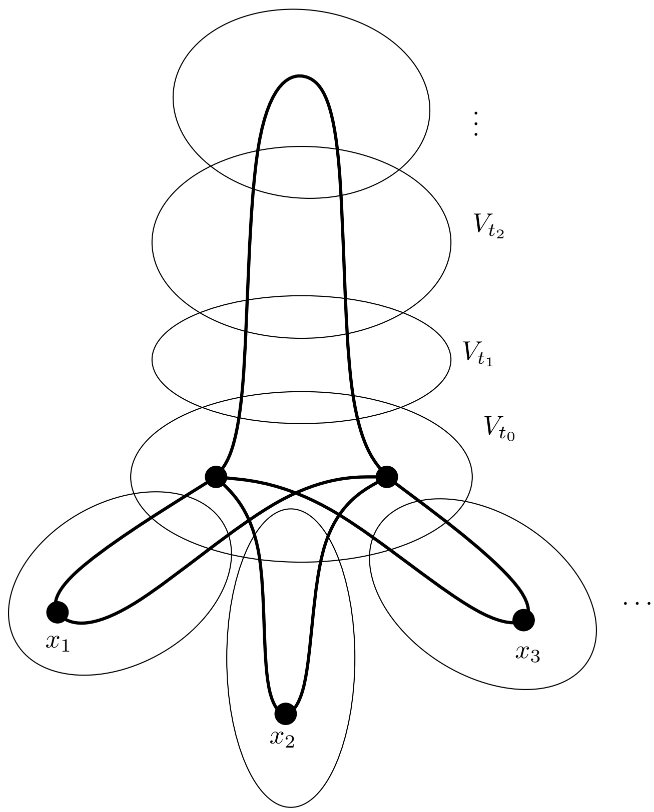


Figure 1: $V_{t_0}$ might grow arbitrarily.

Obviously we made a bad choice here. If we use the path containing $x_1$ for every $V_{t_i}$ we don't need to enlarge $V_{t_0}$ arbitrarily often. This is the idea of the following definition: If we already know a path connecting two vertices $a$ and $b$, then we can reuse it whenever we have a path going through $a$ and $b$.

**Definition 2.1** (navigational path-system (short: navi))**.** Let $G$ be a connected graph and $K \subseteq [V(G)]^{\leq 2}$ a subset of the set of all at most 2-element subsets of $V(G)$. A system $\mathcal{N} := (P_{xy})_{\{x,y\} \in K}$ of $x$–$y$ paths is called *sub-navi*, if for every path $P_{xy}$ in $\mathcal{N}$ and for any two vertices $a$, $b$ on that path $\{a, b\}$ is in $K$ and $P_{ab} = aP_{xy}b$.

A *navi* is a sub-navi satisfying $K = [V(G)]^{\leq 2}$.

If $\mathcal{D} := (T, (V_t)_{t \in T})$ is a tree-decomposition of $G$, then a sub-navi satisfying $[V_t]^{\leq 2} \subseteq K \; \forall \, t \in T$ is called a $\mathcal{D}$-*navi*.

A sub-navi is called *geodesic* if for all $x, y \in K$ the length of $P_{xy}$ is $d_G(x, y)$.

The length of a longest path used in a sub-navi is called the *length* of the sub-navi $l(\mathcal{N}) := max_{\{x,y\}\in K} \|P_{xy}\|$.

A navi knows some connection between every two vertices. A sub-navi might not know all connections, but the known ones are stored in $K$. If a sub-navi knows a path connecting $x$ and $y$, then it knows the connections of all vertex-pairs on that path (they are induced by the original $x$–$y$ path). A $\mathcal{D}$-navi knows the connection of two vertices if they are in a common part of the tree-decomposition $\mathcal{D}$. A geodesic navi does not only know some path between the vertices but a shortest possible. Note that $P_{xy}$ stands for $P_{\{x,y\}}$, so $P_{xy}$ is $P_{yx}$ and in the case of $x = y$ the path $P_{xy}$ is trivial. Let us now see how a navi helps making a tree-decomposition connected:

**Theorem 2.2.** *Let $G$ be a connected graph, $\mathcal{D} = (T, (V_t)_{t\in T})$ a tree-decomposition of $G$ of width $w$ and $\mathcal{N} = (P_{xy})_{\{x,y\}\in K}$ a $\mathcal{D}$-navi of $G$. Define $W_t := \bigcup_{\{x,y\}\in[V_t]^{\le 2}} V(P_{xy})$ for all $t \in T$. Then $(T, (W_t)_{t\in T})$ is a connected tree-decomposition of $G$ of width $\le w + \binom{w+1}{2} \cdot (l(\mathcal{N}) - 1)$.*

*Proof.* Since $\mathcal{N}$ is a $\mathcal{D}$-navi, all $W_t$ are definined. $V_t$ is a subset of $W_t$ for all $t \in T$ because $P_{xy}$ contains $x$ and $y$. So (T1) and (T2) are easy to see. For (T3) let $t_1$, $t_2$ and $t_3$ be distinct vertices of $T$ with $t_2 \in t_1Tt_3$ and let $s$ be in $W_{t_1} \cap W_{t_3}$. We need to show $s \in W_{t_2}$: According to the definition of $W_t$ there must be some $x_1$ and $y_1 \in V_{t_1}$ and some $x_3$ and $y_3 \in V_{t_3}$ such that $s \in P_{x_1y_1}$ and $s \in P_{x_3y_3}$. The set $V_{t_2}$ separates $V_{t_1}$ from $V_{t_3}$, in particular, $V_{t_2}$ is an $\{x_1, y_1\}$–$\{x_3, y_3\}$ separator. If $s \in V_{t_2}$, then $s$ is in $W_{t_2}$ too, as required. So $V_{t_2}$ now has to be a separator without using $s$. This is only possible if $s$ is separated by $V_{t_2}$ from at least one of the sets $\{x_1, y_1\}$ or $\{x_3, y_3\}$ (say $\{x_1, y_1\}$), since otherwise there would be an $\{x_1, y_1\}$–$\{x_3, y_3\}$ path in the union of the $\{x_1, y_1\}$–$s$ path and the $s$–$\{x_3, y_3\}$ path avoiding $V_{t_2}$. Hence there have to be two vertices $x_2$ and $y_2$ in $V_{t_2}$ such that $x_2 \in x_1P_{x_1y_1}s$ and $y_2 \in sP_{x_1y_1}y_1$. By definition of sub-navi $P_{x_2y_2} = x_2P_{x_1y_1}y_2$ and therefore $s \in V(P_{x_2y_2}) \subseteq W_{t_2}$.

All $W_t$ are connected and their size is bounded by "size of $V_t$ + all vertices added". Every $P_{xy}$ has at most $l(\mathcal{N}) - 1$ vertices besides $x$ and $y$ and at most $\binom{w+1}{2}$ of those paths $P_{xy}$ have been added. □

In order to construct a connected tree-decomposition of small width we need to search a $\mathcal{D}$-navi of small length, which is achieved by a geodesic navi. The existence of an arbitrary navi is easy to show, because a spanning tree gives rise to a navi. A bit more surprising is that it is always possible to find a geodesic navi.

**Theorem 2.3.** *Every connected graph has a geodesic navi.*

*Proof.* Let $G = (V, E)$ be the connected graph with a fixed linear order of the vertex set. The set of characteristic vectors of geodesic paths in $G$ is by lexicographical order again linearly ordered. Since there are no two different geodesic paths on the same vertex set, there is a 1-1-correspondence between the characteristic vectors of geodesic paths and the paths themselves. So the set of geodesic paths is ordered lexicographically too. Note that there is a geodesic path between any two vertices as $G$ is connected. Hence for every two vertices $x$ and $y$ in $G$ there is exactly one minimal geodesic $x$–$y$ path. Declare this path to be $P_{xy}$. Then $\mathcal{N} := (P_{xy})_{\{x,y\}\in[V(G)]^{\leq 2}}$ is a path-system consisting of geodesic paths.

Assume that $\mathcal{N}$ is not a navi. Then there are two vertices $x$ and $y$ in $V$ and $a, b \in P_{xy}$ such that $Q_{ab} := aP_{xy}b \neq P_{ab}$.

Observe that $Q_{ab}$ is a geodesic $a$–$b$ path and $Q_{xy} := xP_{xy}aP_{ab}bP_{xy}y$ is a geodesic $x$–$y$ path. So they were considered when declaring $P_{xy}$ and $P_{ab}$, but were not chosen because $P_{ab} < Q_{ab}$ and $P_{xy} < Q_{xy}$. Since $P_{xy} - Q_{ab} = Q_{xy} - P_{ab}$, we can extend $Q_{ab}$ to $P_{xy}$ and $P_{ab}$ to $Q_{xy}$ using the same paths (i.e. without changing the lexicographical ordering). This is a contradiction, so $\mathcal{N}$ is in fact a geodesic navi of $G$. □

Given a tree-decomposition $\mathcal{D} = (T, (V_t)_{t\in T})$ and a geodesic navi $\mathcal{N} = (P_{xy})_{\{x,y\}\in[V(G)]^{\leq 2}}$ we can define a geodesic $\mathcal{D}$-navi by collecting only the needed paths: $\mathcal{N}_{\mathcal{D}} := (P_{xy})_{\{x,y\}\in K_{\mathcal{D}}}$ with $K_{\mathcal{D}} := \bigcup_{t\in T}\bigcup_{\{x,y\}\in V_t}[P_{xy}]^{\leq 2}$. The length of this navi $\mathcal{N}_{\mathcal{D}}$ is bounded by the maximal distance of two vertices, which live inside a common part of $\mathcal{D}$. The task has now changed into finding a tree-decomposition of width $tw(G)$ such that two vertices living inside a common part have a distance bounded by the tree-width of $G$ and the length of a longest geodesic cycle.

## 3 Atomic tree-decompositions

In a contradiction proof it might be useful to not be able to refine a tree-decomposition. Technically this can be achieved by considering the descending ordered sequences of part-sizes of the possible tree-decompositions of the graph. A lexicographically minimal such sequence shall be called *atomic*. An equivalent version of the same idea that shortens the argument can be found in [2] (in the proof of Theorem 3 on page 3):

**Definition 3.1** (atomic tree-decomposition as in [2])**.** Let $G$ be a graph and $n := |G|$. Let the *fatness* of a tree-decomposition of $G$ be the $n$-tuple $(a_0, \dots, a_n)$, where $a_h$ denotes the number of parts that have exactly $n-h$ vertices. A tree-decomposition of lexicographically minimal fatness is called an *atomic* tree-decomposition.

Since there always exists a tree-decomposition that has no part of size $> tw(G)+1$ it is clear that an atomic tree-decomposition has width $tw(G)$.

## 3.1 Rearranging tree-decompositions

Let us introduce some constructions that will reveal useful properties of atomic tree-decompositions. One possible way of rearranging a tree-decomposition is contracting an edge in its tree:

**Lemma 3.2.** *Let $G$ be a graph, $\mathcal{D} = (T, (V_t)_{t \in T})$ a tree-decomposition of $G$ and $e = rs$ an edge of $T$. Define $T' := T/e$, $W_t := V_t \; \forall t \in T - \{r, s\}$ and $W_{t_e} := V_r \cup V_s$. Then $\mathcal{D}' := (T', (W_t)_{t \in T'})$ is a tree-decomposition of $G$.*

*Proof.* (T1) and (T2): Every vertex and every edge of $G$ was inside one $V_t$, which now lives inside a $W_t$. (T3): Let $t_1$, $t_2$ and $t_3$ be distinct vertices of $T'$ with $t_2 \in t_1 T' t_3$. Consider the contracted vertex $t_e$: If $t_e \notin \{t_1, t_2, t_3\}$, then $W_{t_1} \cap W_{t_3} = V_{t_1} \cap V_{t_3} \subseteq V_{t_2} = W_{t_2}$. If $t_e = t_2$, then either $r$ or $s$ has to be on the path $t_1 T t_3$, say $r$. Since $\mathcal{D}$ is a tree-decomposition $W_{t_1} \cap W_{t_3} = V_{t_1} \cap V_{t_3} \subseteq V_r \subseteq W_{t_2}$ follows. In the case $t_e = t_1$ (and analog $t_e = t_3$) we know $t_2 \in rTt_3$ and $t_2 \in sTt_3$, which implies $V_{t_2} \subseteq V_r \cap V_{t_3}$ and $V_{t_2} \subseteq V_s \cap V_{t_3}$. By taking the union on both sides we get $(W_{t_2} =) V_{t_2} \subseteq (V_r \cap V_{t_3}) \cup (V_s \cap V_{t_3}) = (V_r \cup V_s) \cap V_{t_3} = W_{t_1} \cap W_{t_3}$, completing the proof of (T3). □

For atomic tree-decompositions this means, that parts are not contained in each other:

**Corollary 3.3.** *Let $G$ be a graph and $\mathcal{D} := (T, (V_t)_{t \in T})$ an atomic tree-decomposition of $G$, then $V_r \nsubseteq V_s$ for all distinct $r, s \in T$.*

*Proof.* Assume there are two distinct vertices $r$ and $s$ in $T$ with $V_r \subseteq V_s$. By (T3) every vertex from $V_r \cap V_s$ (which is $V_r$ by assumption) is contained in every part on the path $rTs$. Especially the neighbor $t_0$ of $r$ in $rTs$ satisfies $V_r \subseteq V_{t_0}$. Contract the edge $e = rt_0$ in the tree-decomposition $\mathcal{D}$ using Lemma 3.2 and note that the contracted part $W_{t_e}$ equals $V_r \cup V_{t_0} = V_{t_0}$. This means that $\mathcal{D}'$ has exactly one part of size $|V_r|$ less than $\mathcal{D}$ (the other sizes of parts are the same). So $\mathcal{D}'$ has a smaller fatness than the atomic tree-decomposition $\mathcal{D}$, which cannot be. □

Another tool is "separating the components of a subtree-decomposition". In order to formalize this we need some notation:

**Definition 3.4.** Let $G$ be a connected graph, $\mathcal{D} = (T, (V_t)_{t\in T})$ a tree-decomposition of $G$ and $e = st_0 \in E(T)$. Let $T_0$ be the component of $T - e$ containing $t_0$ and $T_s$ the other one (containing $s$). Define $G_0 := G[\bigcup_{t\in T_0} V_t]$, $G_s := G[\bigcup_{t\in T_s} V_t]$ and $X := V_s \cap V_{t_0}$. Let $\mathcal{C} = \{C_1, \dots, C_n\}$ be the set of components of $G_0 - X$ (equivalently, of $G - G_s$) and $N_1, \dots, N_n$ their neighborhoods (in $X$) i.e. $N(C_j) = N_j$, $j = 1, \dots, n$. Let $T_1, \dots, T_n$ be disjoint copies of $T_0$ and $\varphi_i : T_0 \longrightarrow T_i$ be the canonical map, mapping every vertex $t \in T_0$ to its copy in $T_i$.

Define $G_i := V(C_i) \cup N_i$ and $W_{\varphi_i(t)} := V_t \cap G_i$ for $t \in T_0$ and $1 \leq i \leq n$. Set $W_t := V_t$ for $t \in T_s$ and furthermore, $T' := T - T_0 + T_1 + \ldots + T_n + s\varphi_1(t_0) + \ldots + s\varphi_n(t_0)$.

**Lemma 3.5.** *Let the situation of Definition 3.4 be given. Then* $\mathcal{D}' := (T', (W_t)_{t\in T'})$ *is a tree-decomposition of* $G$.

*Proof.* (T2): Let $e = xy \in E(G)$ be an edge of $G$, then one part $V_t$ of $\mathcal{D}$ contains both ends of $e$. If $x$ and $y$ are in $G_s$, then they are in one unchanged $V_t = W_t$ (for some $t \in T_s$). If they are not both in $G_s$, then one of them, say $x$, is in one component $C_i$ of $G - G_s$. Since all the neighbors of $x$, in particular $y$, lie in $C_i$ or in $N_i$, the ends of the edge $e$ are contained in $V_t \cap G_i = W_{\varphi_i(t)}$. This shows (T1) as well.

(T3): Let $t'_1$, $t'_2$ and $t'_3 \in T'$ be given with $t'_2 \in t'_1 T' t'_3$ and let $t_1$, $t_2$ and $t_3$ be their counterparts in $T$. If there is an index $k \in \{1, \dots, n\}$ with $\{t'_1, t'_2, t'_3\} \subseteq T_s \cup T_k \subseteq T'$, then we can find the path $t'_1 T' t'_3$ in a canonical way in $T$:

- If $t'_2$ is in $T_s$, then so is at least one of $t'_1$ and $t'_3$, say $t'_1$. (T3) for $\mathcal{D}$ implies $W_{t'_1} \cap W_{t'_3} = V_{t_1} \cap W_{t'_3} \subseteq V_{t_1} \cap V_{t_3} \subseteq V_{t_2} = W_{t'_2}$ as desired.
- If, on the other side, $t'_2 \in T_k$, then so is at least one of $t'_1$ and $t'_3$, say $t'_3$. This implies $W_{t'_1} \cap W_{t'_3} \subseteq V_{t_1} \cap (V_{t_3} \cap G_k) = (V_{t_1} \cap V_{t_3}) \cap G_k \subseteq V_{t_2} \cap G_k = W_{t'_2}$ as desired.

In the other case there are distinct indices $k, l \in \{1, \dots, n\}$ such that $t'_1 \in T_k$

and $t'_3 \in T_l$. Because the $N_i$ are disjoint from the $C_i$ we get the inclusion:

$$\begin{aligned}
&W_{t'_1} \cap W_{t'_3} \\
&= (V_{\varphi_k^{-1}(t_1)} \cap G_k) \cap (V_{\varphi_l^{-1}(t_3)} \cap G_l) \\
&\subseteq G_k \cap G_l \\
&= (V(C_k) \cup N_k) \cap (V(C_l) \cup N_l) \\
&= \underbrace{(V(C_k) \cap V(C_l))}_{=\emptyset} \cup \underbrace{(V(C_k) \cap N_l)}_{=\emptyset} \cup \underbrace{(N_k \cap V(C_l))}_{=\emptyset} \cup \underbrace{(N_k \cap N_l)}_{\subseteq X} \\
&\subseteq X
\end{aligned}$$

- If $t'_2 \in T_s$ (which means $t'_2 = s$), then $W_{t'_1} \cap W_{t'_3} \subseteq X \subseteq V_s = W_{t'_2}$.
- If $t'_2 \notin T_s$, then it is without loss of generality in $sT't'_1$ (the case $t'_2 \in sT't'_3$ is analog). Consider the vertices $s$, $t'_2$ and $t'_1$ and use the fact, that they are all in $T_s \cup T_k$. We therefore already know that $W_s \cap W_{t'_1} \subseteq W_{t'_2}$ implying $W_{t'_1} \cap W_{t'_3} \subseteq X(\cap W_{t'_1}) \subseteq W_s \cap W_{t'_1} \subseteq W_{t'_2}$

This completes the proof of (T3). So $\mathcal{D}'$ is a tree-decomposition of $G$. □

## 3.2 Properties of atomic tree-decompositions

Given the situation of Definition 3.4, we say that a part $V_t$ with $t \in T_0$ is *split*, if $|V_t \cap G_i| < |V_t| \forall i \in \{1, \ldots, n\}$. Note that there is a $G_i$ containing $V_t$ if and only if $V_t$ is not split: If there is a $G_i$ containing $V_t$, then $V_t \cap G_i$ is $V_t$, which means that $|V_t \cap G_i|$ is not smaller than $|V_t|$ for this special $i$, so $V_t$ is not split. If $V_t$ is not split, then there is an $i \in \{1, \ldots, n\}$ such that $|V_t \cap G_i| = |V_t|$. Since $V_t \cap G_i$ is a subset of $V_t$, they can only have the same size, if $G_i$ contains $V_t$.

**Lemma 3.6.** *Let the situation of Definition 3.4 be given. If a part $V_t$ with $|V_t| > |X|$ is split, then the resulting tree-decomposition $\mathcal{D}'$ has a smaller fatness than $\mathcal{D}$.*

*Proof.* At first let $V_r$ be a part, which is not split (note: $r \in T_0$). As we will see there is at most one $k \in \{1, \ldots, n\}$ such that $\left|W_{\varphi_k(r)}\right| > |X|$: In the case $|V_r| \leq |X|$ we even know $\left|W_{\varphi_i(r)}\right| \leq |X|$ for all $i \in \{1, \ldots, n\}$, since $V_r$ contains every $W_{\varphi_i(r)}$. In the other case there has to be at least one vertex $a$ of $G$ which is in $V_r$ but not in $X$. This vertex is contained in one of the components of $G_0 - X$ and hence in one $G_k$. Since $V_r$ is not split we know that there is a $k$ such that $G_k$ contains $V_r$, hence the intersection of $V_r$ and a $G_i$ with $i \neq k$ is a subset of $X$ and therefore $\left|W_{\varphi_i(r)}\right| \leq |X|$ for all $i \neq k$.

Let $V_r$ now be a part of maximal size that is split (i.e. all the $W_{\varphi_i(r)}$ are smaller than $V_r$). By prerequisites $|V_r| > |X|$, therefore every part of $\mathcal{D}$, which has at least size $|V_r|$ and is not split, induces only one part of its original size and the other induced parts are smaller than $X$. For the comparison of the fatnesses $(a_0, \ldots, a_n)$ of $\mathcal{D}$ and $(a'_0, \ldots, a'_n)$ of $\mathcal{D}'$ this means, that the entries before[1] $a_{n-|V_r|}$ are equal and that $a'_{n-|V_r|}$ is by at least one smaller than $a_{n-|V_r|}$. So $\mathcal{D}'$ has a (lexicographically) smaller fatness than $\mathcal{D}$. □

If a "big"[2] part $V_t$ is split, then the resulting tree-decomposition is smaller than the original one, which therefore was not atomic. So in an atomic tree-decomposition no "big" part is split. In particular the "first part in $G_0$" $V_{t_0}$ is such a big part, because it contains $V_{t_0} \cap V_s = X$ and at least one more vertex in $G - G_s$ (since otherwise $V_{t_0}$ would be a subset of $V_s$, contradicting Corollary 3.3). For justification of the term "atomic" we will show that even "small" parts are not split in an atomic tree-decomposition:

**Lemma 3.7.** *Let the situation of Definition 3.4 be given, where $\mathcal{D}$ is an atomic tree-decomposition. Then $V_t$ is not split for all $t \in T_0$.*

*Proof.* Suppose $\tilde{t}_0$ is a vertex in $T_0$ corresponding to a split part. Let $\tilde{s}$ be the neighbor of $\tilde{t}_0$ on the path $\tilde{t}_0 T s$ and $\tilde{e} := \tilde{s}\tilde{t}_0$. Let $\tilde{T}_0$ be the component of $T - \tilde{e}$ containing $\tilde{t}_0$ and $\tilde{T}_s$ the other one (containing $\tilde{s}$). Define $\tilde{G}_0 := G[\bigcup_{t \in \tilde{T}_0} V_t]$ and $\tilde{G}_s := G[\bigcup_{t \in \tilde{T}_s} V_t]$ furthermore $\tilde{X} := V_{\tilde{s}} \cap V_{\tilde{t}_0}$. Let us first check that every component of $G - \tilde{G}_s$ is contained in a component of $G - G_s$:

By choice of $\tilde{s}$ there is an $\tilde{s}$–$s$ path in $T - \tilde{e}$. Combining this path with another path connecting $s$ with a vertex of $T_s$ we get a path from every vertex of $T_s$ to $\tilde{s}$ in $T - \tilde{e}$, since $T_s$ does not contain $\tilde{t}_0$ and therefore cannot contain $\tilde{e}$. This means that every vertex of $T_s$ lives in the component of $T - \tilde{e}$ which contains $\tilde{s}$. So we have $T_s \subseteq \tilde{T}_s$ which implies $G_s \subseteq \tilde{G}_s$. Every component of $G - \tilde{G}_s$ is disjoint from $\tilde{G}_s \supseteq G_s$. These components are connected and are therefore contained in a maximal connected subset of $G - G_s$. So every component of $G - \tilde{G}_s$ is contained in a component of $G - G_s$. Now we construct another situation as in Definition 3.4 at the edge $\tilde{e}$.

By Lemma 3.6 we now know that $V_{\tilde{t}_0}$, being the "big" part next to $V_{\tilde{s}}$, is not split in this new situation. So there is a component $\tilde{C}$ of $G - \tilde{G}_s$

[1] where the parts of size larger than $|V_r|$ are counted

[2] big means $|V_t| > |X|$

such that $V(\tilde{C}) \cup \tilde{N}$ contains $V_{\tilde{t}_0}$, where $\tilde{N}$ is the neighborhood of $\tilde{C}$. The component $\tilde{C}$ is contained in a component $C$ of $G - G_s$ and therefore $C \cup N$ contains $\tilde{N}$, where $N$ is the neighborhood of $C$. There is an $i \in \{1, \dots, m\}$, such that $V(C) \cup N = G_i$. Now we know $V_{\tilde{t}_0} \subseteq V(\tilde{C}) \cup \tilde{N} \subseteq V(C) \cup N = G_i$. So $V_{\tilde{t}_0}$ is not split even in the original situation. □

After we have seen that there are no split parts in atomic tree-decomposition, we shall now see why this is useful.

**Lemma 3.8.** *Let $G$ be a connected graph, $\mathcal{D} = (T, (V_t)_{t \in T})$ an atomic tree-decomposition of $G$ and $e = st_0 \in E(T)$. Use the notation of Definition 3.4. Then the neighborhood of $C_0$, the component of $G_0 - X$ meeting $V_{t_0}$, is all of $X$.*

*Proof.* Since $\mathcal{D}$ is an atomic tree-decomposition, $V_{t_0}$ is not split (by Lemma 3.7). This means that there is a component $C_i$ such that the corresponding $G_i$ (which is $V(C_i) \cup N_i$) contains $V_{t_0}$ and since $X$ does not contain all of $V_{t_0}$ we get an element $a$ in $V_{t_0} \cap V(C_i)$. If there would be another component meeting $V_{t_0}$ (in $b$), then $V_{t_0}$ would be split, because then every $G_i$ misses at least one of the vertices $a$ or $b$ and therefore every $|V_{t_0} \cap G_i|$ is smaller than $|V_{t_0}|$. Now we are allowed to speak of "*the* component $C_0$ meeting $V_{t_0}$".

As we have seen $V_{t_0}$ is a subset of $V(C_0) \cup N_0$, where $N_0$ is the neighborhood of $C_0$. Since $V_{t_0}$ contains $X$ we know $X \subseteq V_{t_0} \subseteq V(C_0) \cup N_0$. This implies $X \subseteq N_0$ because $X$ is disjoint from the component $C_0$. So every vertex of $X$ is a neighbor of a vertex in $C_0$. □

Given two vertices $u$ and $v$ living inside one common part $V_s$. If there is an edge $st_0$ in $T$ such that both vertices live in $V_{t_0}$ too, then there is (byLemma 3.8) a $u$–$v$ path $P$ (going through the component $C_0$), whose inner vertices are all in $G_0 - X$. Changing the roles of $t_0$ and $s$ we get another $u$–$v$ path $Q$, whose inner vertices are all in $G_s - X$. Combining those paths we get a cycle $C := P \cup Q$ containing $u$ and $v$, which lies "nice" in $G$ (with respect to the tree-decomposition). An even nicer fact is, that the used intersection $X$ always exists, if needed.

**Lemma 3.9.** *Let $\mathcal{D} = (T, (V_t)_{t \in T})$ be an atomic tree-decomposition of a connected graph $G$. If $u$ and $v$ are two vertices living inside a common part $V_s$, then at least one of the following holds:*

- *$uv$ is an edge of $G$.*
- *There is a neighbor $t_0$ of $s$ in $T$, such that $\{u, v\} \subseteq V_s \cap V_{t_0}$*

*Proof.* Assume both statements are false, then there is a part $V_s$ containing two non-adjacent vertices $u$ and $v$, such that for every neighbor $t$ of $s$ either $u$ or $v$ (or both) is missing in $V_s \cap V_t$.

Define a new tree-decomposition $(T', \mathcal{W} = (W_t)_{t \in T'})$ by "de-contracting $V_s$" as follows:

The new tree lives on $V(T') := V(T) - s + t_u + t_v$ where $t_u$ and $t_v$ are two new vertices. Let $N$ be the neighborhood of $s$ in $T$ and $U := \{t \in N : v \notin V_t\}$ the set of neighbors lacking $v$. Let $e$ be the edge $t_u t_v$ then the edge set of $T'$ is $E(T') := E(T - s) + \{tt_u : t \in U\} + \{tt_v : t \in N - U\} + e$. Since the old neighbors of $s$ are distributed among $t_u$ and $t_v$, we know that $T'$ is a tree. Let $W_t := V_t \ \forall \ t \in T - s$, $W_{t_u} := V_s - v$, $W_{t_v} := V_s - u$ and $\mathcal{D}' := (T', (W_t)_{t \in T'})$.

Let $T_u$ be the component of $T' - e$ containing $t_u$ and analog $T_v$ the other one (containing $t_v$), then every part corresponding to a vertex in $T_u$ does not contain $v$ (and vice versa): The parts $W_t$ with $t \in U \cup \{t_u\}$ do not contain $v$ by definition. For the other parts $W_{t'}$ we consider the path $P := sTt'$ in $\mathcal{D}$ and note that it contains a vertex $u'$ of $U$ by construction. If $v$ would be in $W_{t'} = V_{t'}$, then it would be in $V_s \cap V_{t'}$ but not in $V_{u'}$, which is a contradiction. The other statement $u \notin W_t \ \forall t \in T_v$ can be shown in an analog way. Now we will see that $\mathcal{D}'$ is a tree-decomposition of $G$.

(T1) holds, because $V_s = W_{t_u} \cup W_{t_v}$. (T2) holds, because $u$ and $v$ are not adjacent. For (T3) let $t_1$, $t_2$ and $t_3$ be vertices of $T'$ with $t_2 \in t_1 T' t_3 =: P'$. By contraction of $e$ we get a $t_1$–$t_3$ path $P$ in $T$ containing $t_2$ (we identify $t_u$ and $t_v$ in $T'$ with $s$ in $T$ and everything else is unchanged): If $t_2$ is none of $t_u$ and $t_v$, then we know $W_{t_1} \cap W_{t_3} \subseteq V_{t_1} \cap V_{t_3} \subseteq V_{t_2} = W_{t_2}$. If $t_2$ is $t_u$, then $W_{t_1} \cap W_{t_3} \subseteq V_{t_1} \cap V_{t_3} \subseteq V_{t_2} = W_{t_u} \cup \{v\}$. This would only be a problem if $v \in W_{t_1} \cap W_{t_3}$, but in this case both vertices $t_1$ and $t_3$ cannot be in $T_u$. So they are in $T_v$, which means that $t_2$ is not on $P'$. This contradiction shows $W_{t_1} \cap W_{t_3} \subseteq W_{t_2}$. The last case $t_2 = t_v$ is analog.

Hence $\mathcal{D}'$ is a tree-decomposition which has exactly one part of size $|V_s|$ less than $\mathcal{D}$ and two smaller parts are added. So $\mathcal{D}'$ has a (lexicographically) smaller fatness than the atomic $\mathcal{D}$. This contradiction shows that at least one of the statements has to be true. □

## 4 $\mathcal{C}$-Closure

Now that we have a suitable (atomic) tree-decomposition and know how to turn it into a connected tree-decomposition (using a navi), we just have to show that its width is bounded (by a number only depending on the tree-width  and the length of a longest geodesic cycle). The following definition will be a useful tool, because it exhibits the subgraph that will contain the desired path of bounded length:

**Definition 4.1.** Let $G$ be a graph and $\mathcal{C}$ a set of cycles in $G$. Define the $\mathcal{C}$-*Closure* of a vertex-set $X$, to be the union of the cycles in $\mathcal{C}$ meeting $X$. In signs: $\mathcal{C}l(X) := \bigcup_{C\in\mathcal{C}_X} C$ with $\mathcal{C}_X := \{C \in \mathcal{C} : C \cap X \neq \emptyset\}$.

If every $x \in X$ is on a cycle in $\mathcal{C}$ then obviously $X \subseteq \mathcal{C}l(X)$. If, on the other hand, there is a vertex $x$ in $X$ which misses every cycle in $\mathcal{C}$ then $x \notin \mathcal{C}l(X)$, consequently $X \nsubseteq \mathcal{C}l(X)$. If $X \subseteq Y$ then $C_X \subseteq C_Y$ and therefore $\mathcal{C}l(X) \subseteq \mathcal{C}l(Y)$. Since the inclusion $\mathcal{C}l(\mathcal{C}l(X)) \subseteq \mathcal{C}l(X)$ is false in general, the $\mathcal{C}$-Closure is not a closure-operator. The $\mathcal{C}$-closure of a set helps us finding an upper bound for the distance of the vertices in that set.

**Lemma 4.2.** *Let $G$ be a graph and $\mathcal{C}$ a set of cycles in $G$ whose length is bounded by $k$. Let $X \subseteq V(G)$ be a vertex-set with $X \subseteq \mathcal{C}l(X)$. If $\mathcal{C}l(X)$ is connected then every two vertices in $X$ have a distance $\leq k \cdot (|X| - 1)$ in $G$.*

*Proof.* Let us first show that for every bipartition $\{A, B\}$ of $X$ their $\mathcal{C}$-closures meet, i.e. $\mathcal{C}l(A) \cap \mathcal{C}l(B) \neq \emptyset$. Since $X \subseteq \mathcal{C}l(X)$ is equivalent to every vertex in $X$ (which is $A \cup B$) being on a cycle in $\mathcal{C}$, we know $A \subseteq \mathcal{C}l(A)$ and $B \subseteq \mathcal{C}l(B)$. Every edge $xy$ in $\mathcal{C}l(X)$ lies on a cycle $C$ of $\mathcal{C}$ meeting $X$ (in $A$ or $B$ (or both) since $\{A, B\}$ is a partition of $X$). This shows that $x$ and $y$ are in $\mathcal{C}l(A)$ or $\mathcal{C}l(B)$ (or both), so $\mathcal{C}l(X) = \mathcal{C}l(A) \cup \mathcal{C}l(B)$.

Choose two vertices $a \in A$ and $b \in B$ and an $a$–$b$ path $P \subseteq \mathcal{C}l(X)$ (there is one, since $\{a, b\} \subseteq A \cup B = X \subseteq \mathcal{C}l(X)$ and $\mathcal{C}l(X)$ is connected). Consider the the first (i.e. closest to $a$) vertex $y$ in $P$ which is in $\mathcal{C}l(B)$ (there is one, since $b$ is a candidate). In the case that $y$ equals $a$ we have found a vertex in the intersection of $\mathcal{C}l(A)$ and $\mathcal{C}l(B)$. In the other case the predecessor $x$ of $y$ on $P$ has to be in $\mathcal{C}l(A)$. If $\mathcal{C}l(A)$ contains the edge $xy$, then $y$ lies in $\mathcal{C}l(A) \cap \mathcal{C}l(B)$, in the other case the intersection contains $x$.

Now construct an auxiliary tree $T$ on $X$, such that the distance (in $G$) of every pair of vertices that are adjacent in $T$ is bounded by $k$. The construction begins with an arbitrary vertex of $X$ as a single vertex tree $T_0$. If the tree $T_i$ is constructed, we can consider the partition $\{V(T_i), X - V(T_i)\}$. Now we know that their $\mathcal{C}$-closures meet, i.e. there are vertices $x \in V(T_i)$ and $y \in X - V(T_i)$ and intersecting cycles $C_x$ and $C_y$ in $\mathcal{C}$ with $x \in C_x$ and $y \in C_y$. Applying the triangle inequality to $x$, $y$ and a vertex $z$ in the intersection of the cycles $C_x$ and $C_y$, we get an upper bound for the distance between $x$ and $y$:

$$d_G(x,y) \leq d_{C_x}(x,z) + d_{C_y}(z,y) \leq \left\lfloor \frac{k}{2} \right\rfloor + \left\lfloor \frac{k}{2} \right\rfloor = 2\left\lfloor \frac{k}{2} \right\rfloor \leq k$$

In order to get the tree $T_{i+1}$ we add $y$ and the edge $xy$ to $T_i$.

At the end of this iteration we get a tree $T$ (living on all of $X$). For every pair of vertices in $X$ there is a path connecting them in $T$. This path has at most $|V(T)| - 1$ edges and every pair of adjacent vertices has a distance of at most $k$ in $G$. Combining this we get $d_G(x,y) \leq k \cdot (|X| - 1)$ for every two vertices $x$ and $y$ in $X$. □

Now we want to combine the $\mathcal{C}$-closure with the "nice" cycle that we found in Section 3 (before Lemma 3.9). We will use the cycle space, so notations like "generate" or "+" have to be read in the sense of the edge space here (whereas "−" remains set-deletion).

**Lemma 4.3.** *Let $G$ be a graph and $\{G_s, G_0\}$ a separation of $G$ and $X := G_s \cap G_0$ the separator. Let $C = P \cup Q$ be a cycle consisting of two $x$-$y$ paths $P$ and $Q$ such that $P - \{x,y\} \subseteq G_0 - X$ and $Q \subseteq G_s$. Let $\mathcal{C}$ be a set of cycles such that $C$ lies in the subspace they generate.*

*Then there exists an $x$–$y$ path in the $\mathcal{C}$-closure $\mathcal{C}l(X)$ of $X$.*

*Proof.* Let us proof the following statement first:

**Claim.** *Let $G$ be a graph, $P$ an $x$–$y$ path in $G$ and $Z$ an element of the cycle space of $G$. Then there is an $x$–$y$ path in $P + Z$.*

Let $e := xy$ be the (theoretical) edge connecting $x$ and $y$. In the case $e \notin (P + e) + Z$, we know $e \in P + Z$ and hence $e$ is the desired $x$–$y$ path in $P + Z$. In the other case $e \in (P + e) + Z$ there are two possibilities: On the one hand the path $P$ might be just the edge $e$, then $P + e$ is empty. On the other hand the path $P$ might be not that edge $e$, then $P$ (being an $x$–$y$ path) does not even contain $e$ so $P + e = P \cup e$ is a cycle. In both cases $(P + e) + Z$ is an element of the cycle space of $G \cup e$ and therefore a disjoint union of cycles in $G \cup e$. One of these cycles $C'$ has to contain $e$. So $P' := C' - e \subseteq ((P + e) + Z) - e \subseteq (G \cup e) - e \subseteq G$ is the desired $x$–$y$ path in $P + Z$ completing the proof of the claim.

Coming back to the proof of the lemma we write $C$ as a sum of cycles in $\mathcal{C}$, i.e. $C = \sum_{i \in I} C_i$. Divide $I$ into the cycles on the "left", "right" and "middle" of $X$ by $J_0 := \{j \in I : C_j \subseteq G_0 - X\}$, $J_s := \{j \in I : C_j \subseteq G_s - X\}$ and $J := \{j \in I : C_j \cap X \neq \emptyset\}$. Since every cycle is connected and $\{G_s, G_0\}$

is a separation of $G$, every cycle avoiding $X$ lies either in $G_0 - X$ or in $G_s - X$ (not in both). Therefore $\{J_s, J, J_0\}$ is a partition of $I$.

By the claim there is an $x$–$y$ path $P'$ in $P + \sum_{j \in J_0} C_j$ (whose inner vertices have to lie in $G_0 - X$). Since $Q$ and the $J_s$-cycles are separated by $X$ from $P$ and the $J_0$-cycles, adding them is taking the disjoint union. So we have the following inclusion:

$$\begin{array}{rcl} P' \subseteq & P + Q & + \sum_{j \in J_0 \cup J_s} C_j \\ = & C & + \sum_{j \in J_0 \cup J_s} C_j \\ = & \sum_{j \in J_0 \cup J \cup J_s} C_j & + \sum_{j \in J_0 \cup J_s} C_j \\ = & \sum_{j \in J} C_j & + \emptyset \\ \subseteq & \mathcal{C}l(X) & \end{array}$$

The last inclusion holds, because all cycles from $J$ hit $X$ and are therefore contained in $\mathcal{C}l(X)$. So $P'$ is the desired path in the $\mathcal{C}$-closure of $X$. □

Now we have all the tools needed to prove the main theorem:

**Theorem 1.1.** *The connected tree-width of a graph $G$ is bounded above by a function of its tree-width and the maximum length $k$ of its geodesic cycles. Specifically*

$$ctw(G) \leq tw(G) + \binom{tw(G) + 1}{2} \cdot (k \cdot tw(G) - 1).$$

*Proof.* It is easy to see that the upper bound for the connected tree-width holds if the graph is a forest and $k$ is defined to be $> 0$ , so without loss of generality $k > 2$ and $tw(G) > 1$.

It suffices to prove the theorem for 2-connected graphs:

Let $G$ be a (possibly not 2-connected) graph and $\mathcal{B}$ be a (2-connected) block of $G$. Then the tree-width and the maximum length $l$ of geodesic cycles of $\mathcal{B}$ are bounded above by $tw(G)$ and $k$, respectively.

So the "2-connected version" of the theorem yields a connected tree-decomposition of $\mathcal{B}$ (for bridges and isolated vertices take a one vertex tree-decomposition) of width $\leq tw(\mathcal{B}) + \binom{tw(\mathcal{B})+1}{2} \cdot (l \cdot tw(\mathcal{B}) - 1) \leq tw(G) + \binom{tw(G)+1}{2} \cdot (k \cdot tw(G) - 1)$. We can construct a connected tree-decomposition of the whole graph, by adding edges (according to the block structure[3]) to

[3]For each cutvertex $x$ of the graph we choose for every block $\mathcal{B}_i$, that contains $x$, one vertex $t_i$ in the tree (of the tree-decomposition of $\mathcal{B}_i$), such that $V_{t_i}$ contains $x$. Then we add the edges of a (arbitrary) tree in order to connect all the chosen $t_i$ vertices. When this is done, we do the same procedure for the empty cutset (i.e. we connect the tree-decompositions of the components of the graph).

the disjoint union of the trees (of the connected tree-decompositions of the blocks of the graph) until we get a tree.

So let $G$ be a 2-connected graph (In particular every vertex and every edge of $G$ lies on a (geodesic) cycle). We know how to construct a connected tree-decomposition of width $\leq tw(G)+\binom{tw(G)+1}{2}\cdot(l(\mathcal{N})-1)$ using a navi and an atomic tree-decomposition $\mathcal{D}$ (Theorem 2.2). Because of the existence of a geodesic navi (Theorem 2.3), the length of the used $\mathcal{D}$-navi is bounded by the maximum distance of two vertices living in a common part of the used tree-decomposition.

Let $\mathcal{C}$ be the set of all geodesic cycles of $G$ and $V_s$ be a part of $\mathcal{D}$. If we show that $\mathcal{C}l(V_s)$ is connected, then we know (by Lemma 4.2), that every two vertices in $V_s$ have a distance of at most $k\cdot(|V_s|-1)\leq k\cdot tw(G)$ in $G$. So let $u$ and $v$ be two vertices in $V_s$. By Lemma 3.9 there is either the edge $uv$ (which is then contained in $\mathcal{C}l(V_s)$) or there is a neighbor $t_0$ of $s$ in $T$, such that $u$ and $v$ are contained in the intersection $X := V_s \cap V_{t_0}$. In this case a corollary of Lemma 3.8 is the existence of two $u$–$v$ paths $P$ and $Q$, that form a cycle $C = P\cup Q$ such that $P-\{x,y\}\subseteq G_0-X$ and $Q\subseteq G_s$ ($G_0$ and $G_s$ are defined as in Definition 3.4 and form a separation of $G$). Since $C$, being a cycle, lies in the cycle space which is generated by the geodesic cycles of $G$ (see exercise 32 of chapter 1 in [1]), we can apply Lemma 4.3 and get a $u$–$v$ path in $\mathcal{C}l(X)\subseteq\mathcal{C}l(V_s)$. So for every two vertices of $V_s$ there is a path in $\mathcal{C}l(V_s)$ connecting them. Since the other vertices of $\mathcal{C}l(V_s)$ lie on cycles which hit $V_s$, the $\mathcal{C}$-closure of $V_s$ is connected, as required.

Combining all these pieces, we have shown that $G$ has a connected tree-width of at most $tw(G)+\binom{tw(G)+1}{2}\cdot(k\cdot tw(G)-1)$. □

# 5 Duality

## 5.1 Brambles

A useful tool for determining the tree-width of an unknown graph is a bramble:

If we know a tree-decomposition of width $k$, then we know that the tree-width of $G$ is $\leq k$, but we don't know how much smaller the tree-width is. If we additionally know a bramble of order $k+1$, then (by tree-width duality theorem) the tree-width has to be $\geq k$, hence it equals $k$.

**Definition 5.1.** Two vertex sets are touching if they either intersect or if there is an edge from one to the other. A *bramble* is a set of pairwise touching connected vertex sets. A *cover* of the bramble is a vertex set which intersects every set of the bramble. The *order* of the bramble is the smallest size that a cover of the bramble may have.

The *connected order* of the bramble is the smallest possible size of a connected vertex set covering it.

The tree-width duality theorem says that the only reason for large tree-width is a bramble of large order: It is a reason, because if the graph contains a bramble of large order ($> k$), then it has large tree-width ($\geq k$). And it is the only one, since if it is gone (no bramble of order $> k$), then the tree-width is small ($< k$), so there can be no other reason which rises the tree-width.

### 5.2 Making it connected

The obvious thing to try is finding a "connected tree-width duality theorem", i.e. write a "connected" in front of "tree-width" and see what fits on the bramble side. The natural guess is the connected order:

**Conjecture 1.2.** *Let $k \geq 0$ be an integer. A graph has connected tree-width $\geq k$ if and only if it contains a bramble of connected order $> k$.*

The backward-direction is an easy corollary of the (easy part of the) proof of the tree-width duality theorem, because this direction is shown by the following claim:

**Claim** (from the proof of theorem 12.3.9. in [1])**.** *Given a bramble $\mathcal{B}$ and a tree-decomposition $\mathcal{D}$, then there is a part of $\mathcal{D}$ which covers $\mathcal{B}$.*

This direction can be used to determine the connected tree-width of a cycle (for example):

**Example 5.2.** For a cycle of length $n$ let $\mathcal{B} := [V(C^n)]_c^{\lfloor \frac{n}{2} \rfloor}$ be the set of all connected subsets of size $\lfloor \frac{n}{2} \rfloor$. Let us show that $\mathcal{B}$ is a bramble of connected order $\lceil \frac{n}{2} \rceil + 1$. After deletion of $X \in \mathcal{B}$ and its neighborhood, there are at most (exactly) $\lceil \frac{n}{2} \rceil - 2$ vertices left. Because those are less than $\lfloor \frac{n}{2} \rfloor$ vertices, there is no element of $\mathcal{B}$ inside this rest (i.e. $X$ touches every other element of $\mathcal{B}$, which is therefore a bramble). After deletion of a connected set of size $\leq \lceil \frac{n}{2} \rceil$, there are at least $\lfloor \frac{n}{2} \rfloor$ connected vertices left (containing a non-covered set of $\mathcal{B}$). One more vertex is sufficient to cover $\mathcal{B}$. So $\mathcal{B}$ is a bramble of connected order $\lceil \frac{n}{2} \rceil + 1$ (i.e. the connected tree-width of a cycle of length $n$ is at least $\lceil \frac{n}{2} \rceil$). On the other hand there is a connected tree-decomposition of that cycle consisting of two connected parts of size $\leq \lceil \frac{n}{2} \rceil + 1$ which cover it (see figure 2 for an example). So the connected tree-width of a cycle of length $n$ is $\lceil \frac{n}{2} \rceil$.

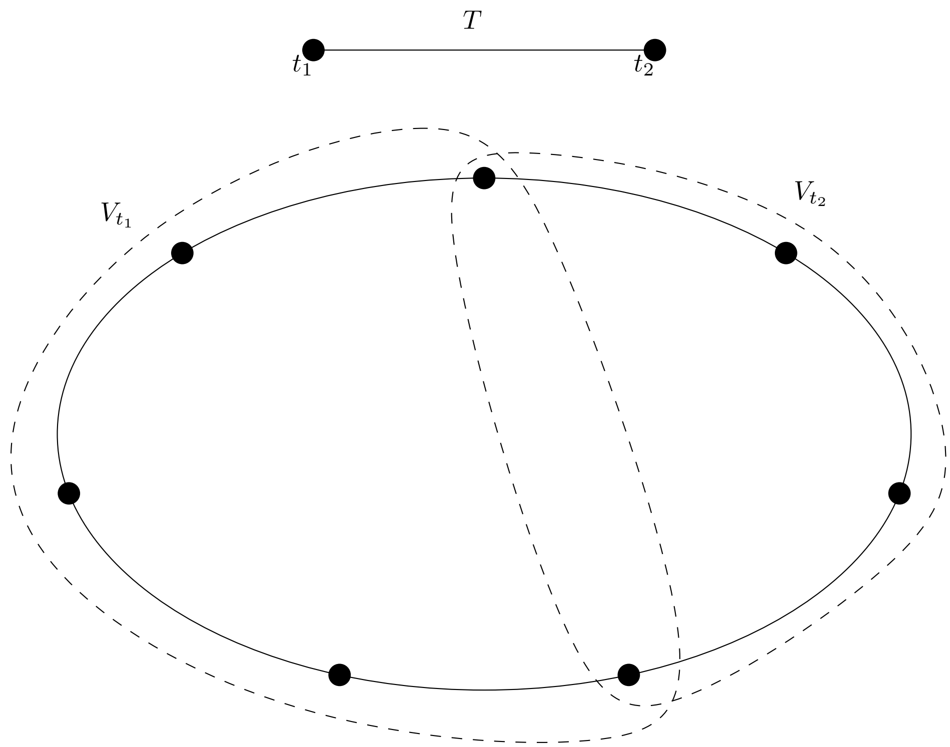


Figure 2: A connected minimum width tree-decomposition of a cycle.

The difficult direction is not that easy to change into the connected version.

By Theorem 1.1 the only two reasons for large connected tree-width are large tree-width and a long geodesic cycle. If we can show that the absence of a bramble of large connected order prevents both these reasons, then we know that the connected tree-width is small (which is the difficult direction of Conjecture 1.2, at least qualitatively). So the next thing to proof is, that a graph with a long geodesic cycle contains a bramble of large connected order (i.e. that a long geodesic cycle really is a reason for large connected tree-width):

**Lemma 5.3.** *If a graph $G$ contains a geodesic cycle $C$ of length $n$, then $G$ has a bramble of connected order $\geq \lceil \frac{n}{2} \rceil + 1$, namely: $\mathcal{B} := [V(C)]_c^{\lfloor \frac{n}{2} \rfloor}$, the set of all connected subsets of $C$ which have size exactly $\lfloor \frac{n}{2} \rfloor$.*

*Proof.* Let $X$ be a connected vertex set in $G$ covering $\mathcal{B}$. We want to show $|X| > \lceil \frac{n}{2} \rceil$:

1. Case: $|X \cap C| = 2$. Then $n$ has to be even and the two vertices $x_0$ and $x_1$ in this intersection $|X \cap C|$ have a distance of $\frac{n}{2}$ in $C$, because otherwise there would be a bramble set not covered by $X$. Since $C$ is geodesic, the distance of $x_0$ and $x_1$ in $G$ is (at least) $\frac{n}{2}$. Because $X$ is connected, there is an $x_0$–$x_1$ path inside $X$, which has at least $\frac{n}{2}+1$ vertices, so $|X| > \lceil \frac{n}{2} \rceil$.

2. Case: $|X \cap C| > 2$. Then there are three vertices $x_0$, $x_1$ and $x_2$ in $X \cap C$. Let $P_i$ be the $x_{i-1}$–$x_{i+1}$ path in $C$ not containing the vertex $x_i$, $i \in \{0, 1, 2\}$ (indices modulo 3). By minimization of the maximal length of these paths we can achieve that $P_i$ is the shorter $x_{i-1}$–$x_{i+1}$ path in $C$: Choose the three vertices in $X \cap C$ such that the maximal size $m$ of the three corresponding paths $P_i$ is minimal. Suppose $|P_1| = m \geq \lfloor \frac{n}{2} \rfloor + 2$, then there is enough space for a bramble set on $P_1$ between $x_0$ and $x_2$. This set is not covered by $x_0$, $x_1$ and $x_2$, so there has to be another vertex in $X$ which takes care of it. Replacing $x_1$ by this vertex we get three new paths $Q_0$, $Q_1$ and $Q_2$ which have all less than $m$ vertices. $Q_0$ and $Q_2$ are proper subpaths of $P_1$ and therefore have less than $m$ vertices. $Q_1$ flipped from $P_1$ to the other side of the cycle, so there are only $n - (\lfloor \frac{n}{2} \rfloor + 2) + 2 = \lceil \frac{n}{2} \rceil < \lfloor \frac{n}{2} \rfloor + 2 \leq m$ vertices left for it. This contradiction to the minimality of $m$ shows that $|P_1| \leq \lfloor \frac{n}{2} \rfloor + 1$ which means that $P_i$ is the shorter of the two $x_{i-1}$–$x_{i+1}$ paths in $C$.

Since $X$ is connected there is an $x_1$–$x_2$ path $P \subseteq X$ and an $x_0$–$P$ path $X_0 \subseteq X$. Let $z := P \cap X_0$, $X_1 := x_1 P z$ and $X_2 := x_2 P z$. So $X_i$ is a path inside $X$ starting at $x_i$ and ending in $z$ (for every $i \in \{0, 1, 2\}$). Since $P_i$ is geodesic and $X_{i-1} \cup X_{i+1}$ is another $x_{i-1}$–$x_{i+1}$ path, we know $|X_{i-1}| + |X_{i+1}| - 1 \geq |P_i|$. Since all the $P_i$ together form the cycle $C$, we know $|P_0| + |P_1| + |P_2| - 3 = n$. Combining this, we get:

$$\begin{aligned} & 2(|X_0| + |X_1| + |X_2|) - 3 \\ =& (|X_1| + |X_2| - 1) + (|X_0| + |X_2| - 1) + (|X_0| + |X_1| - 1) \\ \geq& |P_0| + |P_1| + |P_2| \\ =& n + 3 \end{aligned}$$

Rearranging this, we get:

$$(|X_0| + |X_1| + |X_2|) \geq \frac{n}{2} + 3$$

We can use this to estimate the size of $X$, because all $X_i$ are contained in $X$ and have only $z$ in common:

$$|X| \geq (|X_0| + |X_1| + |X_2|) - 2 \geq \frac{n}{2} + 1 > \left\lceil \frac{n}{2} \right\rceil$$

This shows, that whenever $X$ is a connected set in $G$ which covers $\mathcal{B}$, its size has to be larger than $\lceil \frac{n}{2} \rceil$. So $\mathcal{B}$ is indeed a bramble of connected order $\geq \lceil \frac{n}{2} \rceil + 1$. □

Note that Lemma 5.3 does not naively extend to arbitrary geodesic subgraphs: Let $G$ be the graph indicated in Figure 3 and $H = G - x$ the considered geodesic subgraph. Then there is a bramble of maximal connected order 5 in $H$, namely all 9-element connected subsets of the outer 18-cycle $C$, which has connected order 4 in $G$.

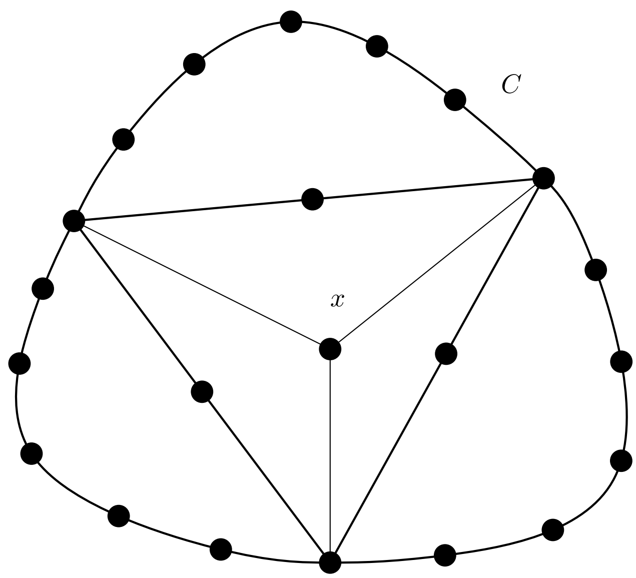


Figure 3: A drawing of the example graph $G$.

It is unknown if there is a graph and a geodesic subgraph such that every maximal connected order bramble of the subgraph has a smaller connected order in the whole graph. In the above example, a bramble of maximal connected order in $H$ whose connected order does not go down in $G$ is the set of all connected 4-element subsets of an 8-cycle in $H$.

Now we can show the qualitative version of the difficult direction of Conjecture 1.2:

**Theorem 1.3.** *Let $k \geq 0$ be an integer. There is a function $g : \mathbb{N} \to \mathbb{N}$, such that any graph with no bramble of connected order $> k$ has connected tree-width $< g(k)$.*

*Proof.* Let $G$ be a graph which has no bramble of connected order $> k$. Since $k \leq 2$ implies that $G$ is a forest, we can assume $k > 2$.

If the graph has a geodesic cycle of length $\geq 2k$, then, by Lemma 5.3, it has a bramble of connected order $\geq \lceil \frac{2k}{2} \rceil + 1 = k + 1$ (which is a contradiction). So there is no geodesic cycle of length $> 2k - 1$ in $G$. The tree-width of $G$ is bounded too, because:

$$\begin{aligned} & G \text{ has no bramble of connected order} > k \\ \Rightarrow\ & G \text{ has no bramble of order} > k \\ \Rightarrow\ & tw(G) < k \end{aligned}$$

The last implication follows from the tree-width duality theorem.

By Theorem 1.1 the connected tree-width of $G$ is bounded by $tw(G) + \binom{tw(G)+1}{2} \cdot ((2k-1) \cdot tw(G) - 1)$ which is smaller than $k + \binom{k+1}{2} \cdot ((2k-1) \cdot k - 1) =: g(k)$, a function only depending on $k$ (note that this function even works for the cases $k \leq 2$). $\square$

## References


[1] R.Diestel. *Graph Theory* (4th edition). Springer-Verlag, 2010. Electronic edition available at: http://diestel-graph-theory.com/.

[2] R.Diestel and P.Bellenbaum. Two proofs concerning tree-decompositions. http://www.math.uni-hamburg.de/home/diestel/papers/LeanTDs.pdf.

[3] P. D. Seymour and Robin Thomas. graph searching, and a min-max theorem for tree-width. *J. Combin. Theory Ser. B 58 (1993), 22-33*, 1989.