Lecture 3: Tree decompositions and the Excluded Grid Theorem

30/05/2022

Lecturer: Marco Bressan

1 Dynamic programming

Computing the size of the largest independent set is NP-hard. Consider however the following toy dynamic program on the graph of Figure 1. We "sweep" G from right to left using a sequence of 6-vertex sets $X_1, \ldots, X_7$. For each i let $X_i^{\text{left}}, X_i^{\text{right}}$ denote the left and right vertices of X_i and let $V_i = \cup_{j=1}^{i} X_i$. For $i = 1$ we list all subsets of $X_i = V_i$. For every $S \subseteq X_i^{\text{left}}$ this yields the size of the largest independent set in V_i whose restriction to X_i^{left} is S. Call this number $\alpha(i, S)$. Now suppose we know $\alpha(i-1, \cdot)$ for some $i > 1$. We list every independent subset A of X_i, and if $S = A \cap X_i^{\text{left}}$, then the largest independent set in V_i whose restriction to X_i^{left} is S has size $|S| + \alpha(i-1, A \cap X_i^{\text{right}})$. Thus we can compute $\alpha(i, S)$ for every subset S of X_i^{left}. This eventually yields us the size of largest independent set in G as $\max_{S \subseteq X_7^{\text{left}}} \alpha(7, S)$.

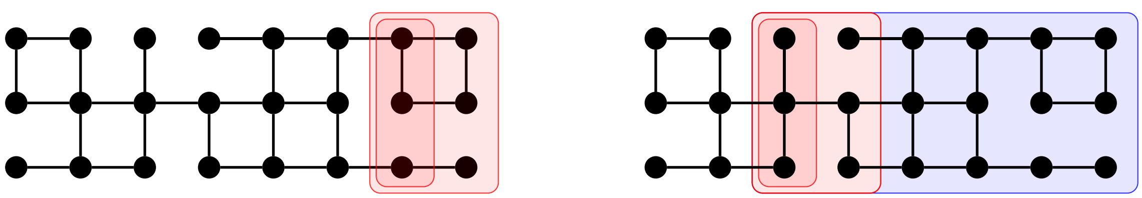

Figure 1: $\alpha(G)$ via dynamic programming. Left: X_1 and X_1^{left}. Right: X_6, X_6^{left} and V_6.

One property exploited by the dynamic program above is that, for every i, there are no edges in G between the two "sides" of G identified by X_i^{left}.

Definition 1.1. Let $G = (V, E)$. A set $X \subseteq V(G)$ is a *(vertex) separator* for $A, B \subseteq V$ if every path between A and B contains a vertex of X.

One can observe that X_i^{left} is a separator for $V_i, V \setminus V_i$. Another way to put it is using separations:

Definition 1.2. A *separation* in a graph $G = (V, E)$ is a pair of sets (A, B) such that $A \cup B = V$ and that G has no edges between $A \setminus B$ and $B \setminus A$. The set $A \cap B$ is called the *separator* and $|A \cap B|$ is called the *order* of (A, B).

In the example above $(V_i, \cup_{j>i} X_j)$ is a separation for every i.

Another crucial property is that these separators form a sequence $X_1, X_2, \ldots$ so that we can check every edge at least once and "in the right order". These properties can be captured formally and generalized from a sequence to a tree.

2 Tree decompositions and treewidth

Definition 2.1. A *tree decomposition* of a graph $G = (V, E)$ is a pair $\mathcal{T} = (T, \{B_t\}_{t \in V(T)})$ where $B_t \subseteq V(G)$ for all $t \in V(T)$ such that:

1. $\cup_{t \in V(T)} B_t = V$
2. $\forall e \in E\ \exists t \in V(T)\ :\ e \subseteq B_t$
3. $B_{t_1} \cap B_{t_2} \subseteq B_t$ for every $t_1, t_2 \in V(T)$ and every t on the unique path between them

The *width* of a tree decomposition $\mathcal{T}$ is

$$w(\mathcal{T}) = \max_{t \in V(T)} |B_t| - 1 \tag{1}$$

The *treewidth* of a graph G is

$$\text{tw}(G) = \min\{w(\mathcal{T})\ :\ \mathcal{T} \text{ tree decomposition of } G\} \tag{2}$$

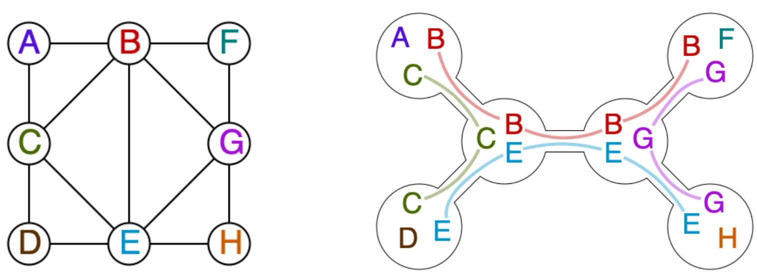

Figure 2: A graph G and its tree decomposition (By David Eppstein - Own work, Public Domain, `https://commons.wikimedia.org/w/index.php?curid=3011976`)

Tree decompositions and treewidth play a central role in graph algorithms, graph minor theory, and algorithmic metatheorems. For example:

Theorem 2.2. *If G is given together with a tree decomposition of width k, then one can find a maximum independent set in G in time $2^k \cdot k^{O(1)} \cdot |V(G)|$.*

Similar results hold for many other NP-hard problems. In fact there exist "metatheorems" saying that, for some function f, every problem of a certain kind (e.g. expressible in a certain logic) can be solved in time $f(k) \cdot V(G)$ whenever $\text{tw}(G) \leq k$.

Exercise 1. *Find a tree decomposition for a tree. What is its width?*

2.1 Properties

Tree decompositions can be thought of as "sequences of separations", but they are actually trees. Let G be any graph and $\mathcal{T} = (T, \{B_t\}_{t \in V(T)})$ be a tree decomposition of G.

Lemma 2.3. *Let $e = \{t_1, t_2\} \in E(T)$, let T_1, T_2 with $t_1 \in V(T_1)$ and $t_2 \in V(T_1)$ be the trees of $T \setminus e$, and let $V_1 = \cup_{t \in V(T_1)} B_t$ and $V_2 = \cup_{t \in V(T_2)} B_t$. Then (V_1, V_2) is a separation in G.*

Proof. We use the properties of Definition 2.1. First, by construction $V_1 \cup V_2 = \cup_{t \in V(T)} B_t$, which by property (1) is $V(G)$. Suppose (V_1, V_2) was not a separation. Thus G contains an edge $\{u, v\}$ with $u \in V_1 \setminus V_2$ and $v \in V_2 \setminus V_1$. Now, by property (2), there is $t \in V(T)$ such that $\{u, v\} \subseteq B_t$. But since $t \in V(T_1) \cup V(T_2)$, at least one of u, v appears in $V_1 \cap V_2$, a contradiction. □

Here is a very useful property of tree decompositions.

Lemma 2.4. *For any $v \in V(G)$ let $T(v) = T[\{t \in V(T) : v \in B_t\}]$. Then $T(v)$ is connected.*

Exercise 2. *Prove lemma Lemma 2.4 and show it is equivalent to property (3) of Definition 2.1.*

2.2 Special cases

Claim 2.5. *If G is a tree then* $\text{tw}(G) = 1$.

Proof. Let G be a tree. Let T be the 1-subdivision of G. For every $t \in V(T)$, if $t = u \in V(G)$ then let $B_t = \{u\}$, and if $t = uv$ for $\{u, v\} \in E(G)$ then let $B_t = \{u, v\}$. It is straightforward to verify that $\mathcal{T} = (T, \{B_t\}_{t \in V(T)})$ satisfies Definition 2.1 and that $w(\mathcal{T}) = 1$. □

Claim 2.6. $\text{tw}(K_n) = n - 1$.

We prove Lemma 2.6 below.

Exercise 3. *The n-by-n* grid $\boxplus_n$ *is defined by:*

$$V(\boxplus_n) = \{(i, j) : i, j \in [n]\} \tag{3}$$

$$E(\boxplus_n) = \{\{(i, j), (i', j')\} : (i, j), (i', j') \in V(\boxplus_n) : |i - i'| + |j - j'| = 1\} \tag{4}$$

Can you find a tree decomposition for $\boxplus_n$ of width $2n - 1$? And of width n?

2.3 Treewidth and minors

Lemma 2.7. *If $H \preceq G$ then* $\text{tw}(H) \leq \text{tw}(G)$.

Proof. Let $\mathcal{T} = (T, \{B_t\})$ be a tree decomposition of G, and consider the following cases:

- if $G' = G \setminus e$, then let $\mathcal{T}' = \mathcal{T}$.
- if $G' = G \setminus v$, then let $\mathcal{T}' = (T, \{B_t \setminus \{v\}\}_{t \in V(T)})$.
- if $G' = G/e$ with $e = \{u, v\}$ then $\mathcal{T}' = (T, \{B'_t\}_{t \in V(T)})$, where $B'_t = B_t$ if $B_t \cap e = \emptyset$ and $B'_t = B_t \setminus e \cup \{uv\}$ otherwise.

One can check that in all cases $\mathcal{T}'$ is a tree decomposition of G', and clearly $w(\mathcal{T}') \leq w(\mathcal{T})$. □

As a consequence:

Corollary 2.8. *For every $k \in \mathbb{N}$ the family $\mathcal{F}_k$ of graphs with treewidth at most k is minor-closed.*

By the graph minor theorem $\mathcal{F}_k$ is characterized by a finite obstruction set $\mathcal{H}$. Unfortunately we do not know $\mathcal{H}$ save for small values of k (e.g., for $k = 1$ we have $\mathcal{H} = \{K_3\}$ since $\mathcal{F}_1$ is precisely the family of acyclic graphs).

2.4 Brambles

Two subsets $X, X' \subseteq V(G)$ *touch* if $X \cap X' \neq \emptyset$ or G has an edge between X and X'.

Definition 2.9. A *bramble* in G is a collection $\beta = \{X_1, \ldots, X_k\}$ of subsets of $V(G)$ such that:

1. $G[X_i]$ is connected for all $i \in [k]$
2. X_i and X_j touch for all $i, j \in [k]$

A set $S \subseteq V(G)$ is a *hitting set* for β if $S \cap X_i \neq \emptyset$ for all $X_i \in \beta$. The *order* $\mathrm{ord}(\beta)$ of β is the size of a smallest hitting set. The *bramble number* of G is

$$\mathrm{bn}(G) = \max\{\mathrm{ord}(\beta) \ : \ \beta \text{ bramble of } G\} \tag{5}$$

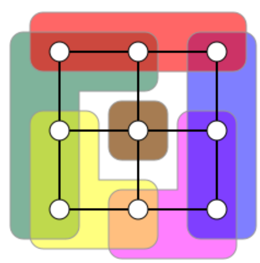

Figure 3: A bramble of order 4 in the 3-by-3 grid graph $\boxplus_3$ (By David Eppstein - Own work, CC0, `https://commons.wikimedia.org/w/index.php?curid=20487288`)

Example 2.10. *$\beta = \{\{v\} : v \in V(G)\}$ is a bramble of order n for $G = K_n$, hence $\mathrm{bn}(K_n) = n$.*

Before proving the next theorem we need some ancillary results.

Lemma 2.11. *If $G[X] \subseteq G$ is connected then $T[X] = T[\{t \in V(T) : X \cap B_t \neq \emptyset\}]$ is connected.*

Proof. By Lemma 2.4 $T[v]$ is connected for every $v \in X$. Now consider any edge $\{u, v\}$ in $G[X]$. By property (2) of Definition 2.1 there is $t \in V(T)$ with $\{u, v\} \subseteq B_t$. But $t \in T[u] \cap T[v]$, thus $T[u] \cup T[v]$ is connected. The proof is completed by iterating over a spanning tree of $G[X]$. □

The following lemma can be proven by induction (we omit the proof).

Lemma 2.12 (Helly property for trees.)**.** *If $T_1, \ldots, T_k$ are subtrees of a tree T, and $V(T_i) \cap V(T_j) \neq \emptyset$ for all $i, j \in [k]$, then $\cap_{i \in [k]} T_i \neq \emptyset$.*

Theorem 2.13. *Every graph G satisfies $\mathrm{tw}(G) \geq \mathrm{bn}(G) - 1$.*

Proof. We use of the *Helly property* for trees (which we do not prove): if $T_1, \ldots, T_k$ are subtrees of a tree T, and $V(T_i) \cap V(T_j) \neq \emptyset$ for all $i, j \in [k]$, then $\cap_{i \in [k]} T_i \neq \emptyset$; that is, there is a vertex contained in every T_i. Let β be a bramble of maximum order in G and let $\mathcal{T} = (T, \{B_t\}_{t \in V(T)})$ be any tree decomposition of G. We prove that some B_t is a hitting set for β.

For any $X \in \beta$, since $G[X]$ is connected, by Lemma 2.11 $T[X]$ is connected. Moreover for every $X, X' \in \beta$, since X and X' touch, the same argument used in the proof of Lemma 2.11 shows that $T(X) \cap T(X') \neq \emptyset$. By Lemma 2.12, there exists $t \in \cap_{X \in \beta} T(X)$. Thus B_t satisfies $B_t \cap X \neq \emptyset$ for all $X \in \beta$. Hence B_t is a hitting set for β, and $|B_t| \geq \mathrm{ord}(\beta)$.

Since β was chosen of maximum order, then $\mathrm{ord}(\beta) = \mathrm{bn}(G)$. Thus $|B_t| \geq \mathrm{bn}(G)$ and $w(\mathcal{T}) \geq \mathrm{bn}(G) - 1$. Since this holds for every $\mathcal{T}$, we conclude that $\mathrm{tw}(G) \geq \mathrm{bn}(G) - 1$. □

Example 2.14. *Let $G = \boxplus_n$. For every $i = 1, \ldots, n$ let $R_i = \{(i, j) : j \in [n]\}$ and $C_i = \{(j, i) : i \in [n]\}$; these are the i-th row and i-th column. Consider:*

$$\beta = \{R_i \cup C_i : i \in [n]\} \tag{6}$$

It is easy to see that β has order n, since any set of less than n vertices misses some row and some column. Hence $\mathrm{bn}(G) \geq n$ and thus $\mathrm{tw}(G) \geq n - 1$. In fact, with a slight modification one can show that $\mathrm{bn}(G) \geq n + 1$ and thus $\mathrm{tw}(G) \geq n$.

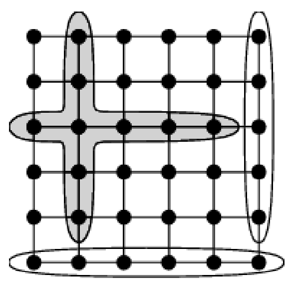

Figure 4: Illustration that $\boxplus_n$ has a bramble of order $n + 1$ and thus $\mathrm{tw}(\boxplus_n) \geq n$. Credit to the authors of *Treewidth Lower Bounds with Brambles*, Algorithmica 51(1):81-98, 2008.

In fact, Robertson and Seymour proved:

Theorem 2.15 (Treewidth Duality Theorem)**.** *Every graph G satisfies $\mathrm{tw}(G) = \mathrm{bn}(G) - 1$.*

2.5 The Excluded Grid Theorem

We conclude with another deep result due to Robertson and Seymour.

Theorem 2.16 (The Excluded Grid Theorem)**.** *There exists a function $f : \mathbb{N} \to \mathbb{N}$ such that, for every $n \in \mathbb{N}$, every graph of treewidth at least $f(n)$ contains $\boxplus_n$ as a minor.*

Hence every graph of treewidth at least (say) 1.000 has (say) a $\boxplus_{10}$ minor, every graph of treewidth at least (say) $1.000.000$ has (say) a $\boxplus_{100}$ minor, and so on. This provides a beautiful "explanation of treewidth": it is attributable to a canonical graph, the grid. Note that this is not true if in place of $\boxplus_n$ one uses, say, K_n (which at first sight may seem an obvious choice).

An equivalent form of Theorem 2.16 is:

Theorem 2.17 (The Excluded Grid Theorem)**.** *There exists a function $f : \mathbb{N} \to \mathbb{N}$ such that every graph that is $\boxplus_n$-minor-free has treewidth less than $f(n)$.*

Moreover, every graph of treewidth k must be $\boxplus_n$-minor-free for every $n > k$, otherwise the treewidth would be larger than k. Thus every graph has a grid minor that "determines" its treewidth. This can be thought of as an approximate version of an obstruction set for graphs of treewidth bounded by k. In particular, an infinite family of graphs $\mathcal{F}$ has unbounded treewidth (i.e. for every $k \geq \mathbb{N}$ it contains a graph of treewidth $\geq k$) if and only if it has unbounded grid minors (i.e. for every $n \geq \mathbb{N}$ it contains a graph with the n-by-n grid as minor). In fact, one usually says that grid minors are obstructions for the treewidth.