ALGORITMICA GRAFURILOR
Săptămâna 14

C. Croitoru

croitoru@info.uaic.ro

FII

January 14, 2015

Ultimul Curs ◡̈ ☺☺☺

OUTLINE

Tree decomposition

Definition

A **tree decomposition** of a graph $G = (V, E)$ is a pair $(T, \{V_t : t \in T\})$, where T is a tree and $\{V_t : t \in T\}$ denotes a family of subsets of the vertices of G, $V_t \subseteq V$ for every node $t \in T$ such that:

- *(Node coverage)* For every $v \in V$, there is some node t in T such that $v \in V_t$.
- *(Edge coverage)* For every $e \in E$, there is some node t in T such that V_t contains both endpoints of e.
- *(Coherence)* Let t_1, t_2, t_3 be three nodes in T such that t_2 lies on the path between t_1 and t_3 in T. Then, if $v \in V$ belongs to both V_{t_1} and V_{t_3} , v must also belong to V_{t_2}.

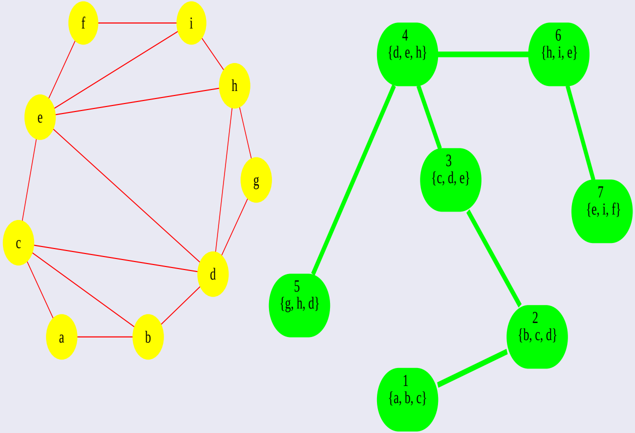
Definition - Comment

- *(Coherence)* Let t_1, t_2, t_3 be three nodes in T s. t. t_2 lies on the path between t_1 and t_3 in T. Then, if $v \in V$ belongs to both V_{t_1} and V_{t_3} , v must also belong to V_{t_2}.
- *(Coherence')* Let t_1, t_2, t_3 be three nodes in T s.t. t_2 lies on the path between t_1 and t_3 in T. Then, $V_{t_1} \cap V_{t_3} \subseteq V_{t_2}$.
- *(Coherence")* For every $x \in V$, the subgraph of T induced by $\{t \in T : x \in V_t\}$ is connected.

The sets V_t are called the **bags** of the tree decomposition.

Tree decomposition

Definition - Example



Graful G

Arborele T

Tree-width

Definition

Let $(T, \{V_t : t \in T\})$ be a tree decomposition of G.
The **width** of tree decomposition $(T, \{V_t : t \in T\})$ is

$$\text{width}(T, \{V_t : t \in T\}) = \max_{t \in T} |V_t| - 1.$$

Definition

The **tree-width** of G, denoted $tw(G)$, is *the minimum width of a tree decomposition of G.*

Tree-width

Observation. $tw(G) = 0$ if and only if $E(G) = \emptyset$.

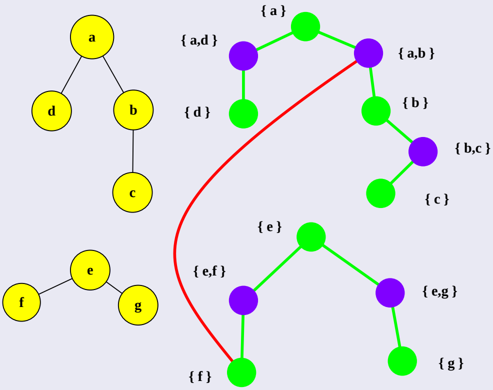
Proposition. *If G is a forest with $E(G) \neq \emptyset$, then $tw(G) = 1$.*
Proof: $tw(G) \geq 1$, by the above observation.
If G is a tree then let T be obtained from G by renaming t_v each vertex $v \in V(G)$ and, after that, inserting on each edge $t_u t_v$ $(uv \in E(G))$ a new vertex t_{uv}.

Set $V_{t_u} = \{u\}$ for all t_u associated to $u \in V(G)$, and $V_{t_{uv}} = \{u, v\}$ for all $t_{uv} \in T$ associated to $uv \in E(G)$.

$(T, \{V_t : t \in T\})$ is a tree decomposition of G with width 1.

A tree decomposition of a forest with k components can be obtained by adding $k - 1$ arbitrary edges to tree decompositions for the components (without creating circuits). $\square$

Tree decomposition with width 1 of a forest



Small tree decompositions

A tree decomposition $(T, \{V_t : t \in T\})$ is **small** if there are no distinct $t_1, t_2 \in T$ such that $V_{t_1} \subseteq V_{t_2}$.

Proposition. *Given a tree decomposition of G, a small tree decomposition of G with the same width can be constructed in polynomial time.*

Proof. Let $(T, \{V_t : t \in T\})$ be a tree decomposition of G with $V_{t_1} \subseteq V_{t_2}$ for $t_1, t_2 \in T$. We can suppose that $t_1 t_2 \in E(T)$ (otherwise, we find adjacent nodes with this property, by considering a path from t_1 to t_2).

Contracting $t_1 t_2$ into a **new node** t_{12} with $V_{t_{12}} = V_{t_2}$, gives a smaller tree decomposition of G. Repeat this reduction until a small tree decomposition is obtained. □

Small tree decompositions

Proposition. *If* $(T, \{V_t : t \in T\})$ *is a small tree decomposition of* G*, then* $|T| \leq |G|$*.*

Proof. By induction over $n = |G|$. If $n = 1$ then $|T| = 1$.

For $n \geq 2$, consider a leaf t_1 of T with neighbor t_2.
$(T - t_1, \{V_t : t \in T - t_1\})$ is a small tree decomposition of $G' = G - (V_{t_1} - V_{t_2})$. Since $V_{t_1} - V_{t_2} \neq \emptyset$, by induction

$$|T| = |T - t_1| + 1 \leq |G'| + 1 \leq |G|.$$

□

Minors

Observations.

- If the graph H is obtained from G by contracting an edge uv into z, then $tw(H) \leq tw(G)$. In a tree decomposition of G, insert z in every bag containing u or v, and then remove u and v from every bag to obtain a tree decomposition of H.
- If H is a subgraph G, then $tw(H) \leq tw(G)$.

Definition. H is a **minor** of a graph G if it can be obtained from G by iteratively deleting and contracting edges.

Corollary. *If H is a minor of a graph G then $tw(H) \leq tw(G)$.*

Tree-width

Let $\mathbf{TW}(k)$ be the class of graphs G such that $tw(G) \leq k$.

Tree-Width (Decision Version)

Input: Graph G and integer k.
Question: $G \in \mathbf{TW}(k)$?

Theorem

Tree-width (decision version) is NP-complete.

Tree-width

Tree-Width is FPT

Lemma. *For every positive integer k, $\mathbf{TW}(k)$ is minor closed.*

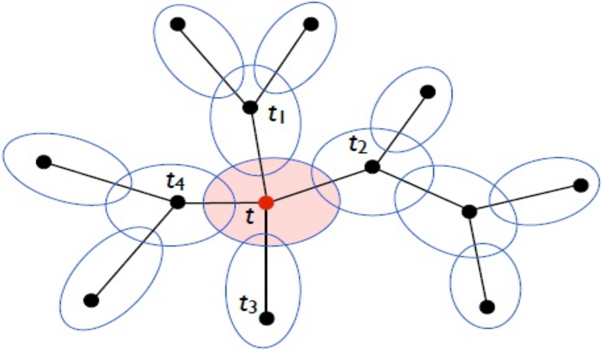
Theorem (Bodlaender). *For every fixed k, the problem of determining whether or not $G \in \mathbf{TW}(k)$ can be solved in $\mathcal{O}(f(k) \cdot n)$ time.*

Notation. Let $(T, \{V_t : t \in T\})$ be a tree decomposition of G. Then, if T' is a subgraph of T, $G_{T'}$ denotes the subgraph of G induced by the set $\bigcup_{t \in T'} V_t$.

Tree decomposition properties

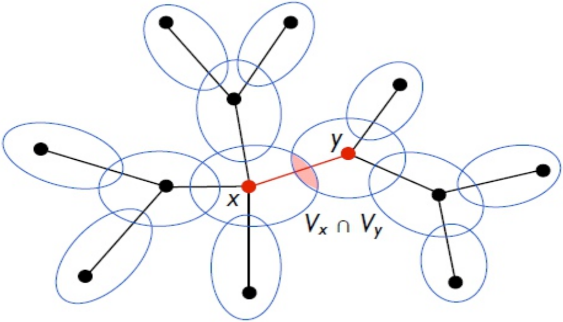
Node Separation Property

Theorem. *Suppose $T - t$ has components $T_1, \ldots, T_d$. Then, the subgraphs $G_{T_1} - V_t, G_{T_2} - V_t, \ldots, G_{T_d} - V_t$ have no nodes in common, and there are no edges between them.*



Edge Separation Property

Theorem. *Let X and Y be the two components of T after the deletion of edge xy. Then, deleting $V_x \cap V_y$ disconnects G into two subgraphs $H_X = G_X - (V_x \cap V_y)$ and $H_Y = G_Y - (V_x \cap V_y)$. That is, H_X and H_Y share no nodes and there is no edge in G with one endpoint in H_X and the other in H_Y.*



Exercises

1. *Let G be a connected graph with $tw(G) = p$. Prove that $|V(G)| = p + 1$ or G has a p-cut.*

2. *Prove that if $tw(G) = 1$, then G is a forest.*

3. *Prove that $tw(P_k \times P_l) = \min(k, l)$.*

4. *Prove that $tw(K_n) = n - 1$.*

Rooted tree decomposition

Definition

A **rooted tree decomposition** of G is a tree decomposition $(T, \{V_t : t \in T\})$ of G, where some node r in T is declared to be the root.

Notations. Let t be a node in a rooted tree decomposition.

- T_t is the subtree of T rooted at t.
- $G[t]$ is the subgraph of G induced by the vertices in $\bigcup_{x \in T_t} V_x$ (i.e. $G[t] = G_{T_t}$).

Vertex coloring

- Recall: A k-vertex coloring (k-coloring) of a graph $G = (V, E)$ is a function $\alpha : V \to \{1, \dots, k\}$ such that for all $uv \in E$, $\alpha(u) \neq \alpha(v)$.
- Let H_1 and H_2 be two subgraphs of G, with k-colorings α_1 and α_2 respectively. α_2 is α_1-compatible if for all $v \in V(H_1) \cap V(H_2)$, $\alpha_1(v) = \alpha_2(v)$.
- Let $(T, \{V_t : t \in T\})$ be a rooted tree decomposition of G. For every $t \in T$ and every k-coloring α of G_t, define

$$\mathbf{Prev}_t(\alpha) = \begin{cases} 1 & \text{if } G[t] \text{ has an } \alpha\text{-compatible } k\text{-coloring } \beta, \\ 0 & \text{otherwise.} \end{cases}$$

Vertex coloring

Proposition. $\mathbf{Prev}_u(\alpha) = 1$ *if and only if for all children v of u, there is an α-compatible coloring β of G_v with* $\mathbf{Prev}_v(\beta) = 1$ *.*

Proof. $\Rightarrow$ If γ is an α-compatible coloring of $G[u]$, since G_v is a subgraph of $G[u]$, the restriction of γ to G_v gives the required coloring β.

$\Leftarrow$ Suppose that u has exactly two children v and w of u, and having α-compatible colorings β and γ respectively (the proof is similar for more children). Since $(T, \{V_t : t \in T\})$ is a tree decomposition, $V(G[v]) \cap V(G[w]) \subseteq V_u$, so β is γ-compatible.
Combining β and γ now gives $\delta : V(G[u]) \to \{1, \ldots, k\}$. Since $(T, \{V_t : t \in T\})$ is a tree decomposition, there are no edges $xy \in E(G)$ with $x \in V(G[v]) - V_u$ and $y \in V(G[w]) - V_u$, so δ is a k-coloring of $G[u]$. $\square$

Vertex coloring

Theorem. *If G, a graph of order n, has a small tree decomposition $(T, \{V_t : t \in T\})$ of width w, we can decide if G is k-colorable in time $k^{w+1} \cdot n^{O(1)}$.*

Proof. Transform $(T, \{V_t : t \in T\})$ in a rooted tree decomposition. For every $v \in T$ and every k-coloring α of G_v, we compute $\mathbf{Prev}_v(\alpha)$: start at the leaves of T, and use the above proposition for the other nodes, in the right order.

$G = G[r]$ is k-colorable iff $\mathbf{Prev}_v(\alpha) = 1$ for some α.

Testing whether α is a G_v coloring and computing $\mathbf{Prev}_v(\alpha)$ can be done in polynomial time $n^{O(1)}$, so the total complexity is mainly determined by the number of candidates for α, which is $k^{|V_v|}$.

Complexity: $|V(T)| \cdot k^{w+1} \cdot n^{O(1)} = k^{w+1} \cdot n^{O(1)}$. □

Applications

Similar approaches (more advanced dynamic programming)

Theorem. *If G, a graph of order n, has a small tree decomposition $(T, \{V_t : t \in T\})$ of width w, the size of a minimum vertex cover of G can be computed in time $2^{w+1} \cdot n^{O(1)}$.*

Theorem. *If G, a vertex-weighted graph of order n, has a small tree decomposition $(T, \{V_t : t \in T\})$ of width w, a maximum weight stable set in G can be computed in time $4^{w+1} \cdot w \cdot n$.*

Anunțuri

1. **Evaluare: ~croitoru/ag/week01.pdf pagina 13**

2. **Pogramare test final: 17,18 ianuarie; atenție la anunțurile de la orar**

   (sunt precizate orele pe grupe și sălile de examen; în situații excepționale, studenții pot veni la alte grupe decât cele programate, dar numai cu acordul meu prealabil).

3. **Seminarul special de vineri seara se suspendă.**

4. ♡ **Succes la examene!**