Dynamic Programming on Trees

Example: Independent Set on $T = (V, E)$ rooted at $r \in V$.

For $v \in V$ let $T_v$ denote the subtree rooted at $v$.

Let $f^+(v)$ be the size of a maximum independent set for $T_v$ that contains $v$. Similarly, $f^-(v)$ is the size of a maximum independent set for $T_v$ that does not contain $v$.

The following algorithm computes a maximum independent set for $T$ in $O(|V|)$ time.

 Traverse $T$ starting from $r$ in post-order. Let $v$ be the current vertex.

- If $v$ is a leaf, let $f^+(v) = 1$ and $f^-(v) = 0$.

- Else let $x_1, \ldots, x_k$ be the children of $v$. Set $f^+(v) = 1 + \sum_{i=1}^{k} f^-(x_i)$ and $f^-(v) = \sum_{i=1}^{k} \max\{f^+(x_i), f^-(x_i)\}$.

Return $\max\{f^+(r), f^-(r)\}$. 
Tree Decompositions

**Definition.** A tree decomposition for a graph $G = (V, E)$ is a pair

$$( \{X_i | i \in I\}, \quad T = (I, F) )$$

such that

- $\bigcup_{i \in I} X_i = V$ (bags cover vertices);
- for each $\{u, v\} \in E$ there is some $i \in I$ s.t. $\{u, v\} \subseteq X_i$ (bags cover edges);
- for all $v \in V$ the set $I_v = \{i \in I | v \in X_i\}$ is connected in $T$ (tree property).

The **width** of a tree decomposition is

$$\max_{i \in I} |X_i| - 1.$$

The **treewidth** of a graph is the minimum width of a tree decomposition for it.

**Example.** Trees have treewidth 1.
Basic Observations

**Observation.** For any graph $G = (V, E)$ a single bag containing $V$ forms a tree decomposition of width $n - 1$.

We are interested in tree decompositions of small width, which certify that the graph is in some way “tree-like”.

Denote the treewidth of a graph $G$ by $\text{tw}(G)$.

**Proposition.** $\text{tw}(H) \leq \text{tw}(G)$ for any subgraph $H$ of a graph $G$.

**Proposition.** If a graph $G = (V, E)$ has two components $A$ and $B$ with $A \cup B = V$ then $\text{tw}(G) = \max\{\text{tw}(A), \text{tw}(B)\}$.
Treewidth of cliques and grids

**Lemma.** Let \( \{X_i \mid i \in I\}, T = (I, F) \) be a tree decomposition for \( G = (V, E) \). For any clique \( G[W], W \subseteq V \), there is an \( i \in I \) such that \( W \subseteq X_i \).

**Proof.** Root \( T \) arbitrarily. For \( w \in W \) let \( r_w \) denote the bag containing \( w \) with minimum height. Then the bag from \( \{r_w \mid w \in W\} \) with maximum height contains \( W \). \( \square \)

In particular, the treewidth of \( K_n \) is \( n - 1 \).

The \( n \times n \)-grid on \( \{(i, j) \mid 1 \leq i, j \leq n\} \) has treewidth \( \leq n \): Consider the path on

\[
X_{n(i-1)+j} = \{(i, k) \mid j \leq k \leq n\} \cup \{(i+1, k) \mid 1 \leq k \leq j\},
\]

\( 1 \leq i \leq n - 1, \ 1 \leq j \leq n. \)
How many vertices are needed in T?

**Definition.** A tree decomposition \( \{X_i \mid i \in I\}, T = (I, F) \) of width \( k \) is **smooth** if

- \( |X_i| = k + 1 \) for all \( i \in I \);
- \( |X_i \cap X_j| = k \) for all \( \{i, j\} \in F \).

**Proposition.** For any graph with treewidth \( k \) there exists a smooth tree decomposition of width \( k \).

**Lemma.** If \( (X, T = (I, F)) \) is a smooth tree decomposition of width \( k \) for \( G = (V, E) \) then \( |I| = |V| - k \).

In particular, \( n(T) \leq n(G) \).
Lemma. A graph $G = (V, E)$ of treewidth at most $k$ has at most $k|V| - \binom{k+1}{2}$ edges.

Proof. Induction on $|V|$. Base case is $|V| = k+1$. Consider a smooth tree decomposition $(\{X_i | i \in I\}, T = (I, F))$ for $G$ and a leaf $i$ of $T$. Then there is a unique vertex $v \in X_i$ that does not belong to any other $X_j$, $j \neq i$. Clearly $\deg_G(v) \leq k$. Removing $i$ from $T$ yields a tree decomposition for $G[V \setminus \{v\}]$. 

Corollary. A graph has treewidth at most one if and only if it is a forest.
Lemma. Let $\{X_i \mid i \in I\}, T = (I, F)$ be a tree decomposition for a connected graph $G = (V, E)$ such that $X_i \not\subseteq X_j$ for all $i, j \in I$. Then

a) $X_i \cap X_j$ is a cut in $G$ for any $\{i, j\} \in F$;

b) $X_i$ is a cut in $G$ for any $i \in I$ that is not a leaf in $T$.

Remark. It is possible to adapt any tree decomposition in $O(|I|)$ time to fulfill the non-containment condition without changing its width.


Treewidth and Separators

**Theorem.** From a given tree decomposition $(\{X_i \mid i \in I\}, T = (I, F))$ of width $k$ for a graph $G = (V, E)$ one can find a $(k + 1, \frac{1}{2})$-separator for $G$ in $O(|I|)$ time.

**Proof.** Root $T$ arbitrarily and define a weight function $w$ on $I$ by $w(i) = |X_i \setminus X_{\text{parent}(i)}|$. Each $v \in V$ is counted exactly once (bags containing $v$ are connected).

Therefore, $\sum_{i \in I} w(i) = |V|$. By the Separator Theorem for (weighted) trees we obtain a $(1, \frac{1}{2})$-separator $s$ for $T$.

Removing $X_s$ disconnects $G$ where

- any $v \in V \setminus X_s$ can appear in at most one subtree (otw, it would also appear in $X_s$ by connectivity);

- each subtree defines at least one component (no edge between subtrees);

- each subtree (and hence component) consists of at most $\frac{n}{2}$ vertices. □
Dynamic Programming on Graphs of Treewidth at most $k$

**Given:** $G = (V, E)$ and a tree decomposition $(\{X_i | i \in I\}, T = (I, F))$ of width $\leq k$ for $G$.

The algorithm below computes a maximum independent set for $G$ in $O(k^2 4^k |V|)$ time.

Pick an arbitrary root $r \in I$ and for $i \in I$ let

$$V_i = \bigcup_{j \in T_i} X_j$$

where $T_i$ denotes the subtree rooted at $i$.

For $U \subseteq X_i$ let $f^U(i)$ be the size of a maximum independent subset of $V_i$ whose intersection with $X_i$ is exactly $U$.

- Traverse $T$ starting from $r$ in post-order. Let $i$ be the current vertex.

- If $i$ is a leaf, for every $U \subseteq X_i$ let
  $$f^U(i) = |U|, \text{ if } U \text{ is independent in } G \text{ and } f^U(i) = -\infty, \text{ otherwise.}$$

- Else let $c_1, \ldots, c_\ell$ be the children of $i$. Set
  $$f^U(i) = |U| + \sum_{j=1}^\ell \max \left\{ f^W(c_j) \mid W \subseteq X_{c_j} \setminus U \land W \cup U \text{ is independent} \right\}.$$
Closure Properties

**Proposition.** Graphs of treewidth at most $k$ are closed under taking minors.

**Proof.** Removal of edges and isolated vertices are trivial. When contracting an edge $\{u, v\}$, replace all occurrences of $u$ in any bag by $v$. 

By Robertson-Seymour there is hence a finite set of forbidden minors. But they are not known, except for small $k$.

- $k = 0$: $K_2$.
- $k = 1$: $K_3$.
- $k = 2$: $K_4$.
- $k = 3$: $K_5$, $K_{2,2,2}$.
- $k = 4$: more than 75...
Computing treewidth

**Theorem** [Arnborg, Corneil, Proskurowski ’87]. Deciding whether the treewidth of a given graph is at most $k$ is NP-complete.

**Theorem** [Bodlaender ’96]. For any $k \in \mathbb{N}$ there exists a linear time algorithm to test whether a given graph has treewidth at most $k$ and—if so—output a corresponding tree decomposition.

(The runtime is exponential in $k^3$.)

**Open.** Can the treewidth be computed in polynomial time for planar graphs?
Not everything is easy for bounded treewidth...

**Theorem [Nishizeki, Vygen, Zhou '01].** Edge-disjoint paths is NP-complete for graphs of treewidth 2.

(But trivial for trees and polynomial for outerplanar graphs.)

Given a graph $G = (V, E)$ and $\{s_i, t_i\} \in \binom{V}{2}$, $1 \leq i \leq k$, find $k$ edge disjoint paths $P_i$ such that $P_i$ connects $s_i$ and $t_i$.

**Theorem [McDiarmid/Reed '01].** Weighted coloring is NP-hard for graphs of treewidth 3.

(But trivial for bipartite graphs → trees.)

Given a graph $G = (V, E)$ and $w : E \to \mathbb{N}$, a weighted $k$-coloring is a function $c : V \to [k]$ such that $|c(u) - c(v)| \geq w(e)$ for all $\{u, v\} \in E$. 
Cops and robber

In the omniscient cops and robber game, \( k \) cops each occupy a vertex of a graph in which a robber moves trying to escape capture. The robber moves along edges “at infinite speed”, the cops move by helicopter.

**Definition.** Given a graph \( G = (V, E) \) and \( k \in \mathbb{N} \), a position in the \( k \) cops and robber game on \( G \) is a pair \((C, r)\), where \( C \in \binom{V}{k} \) (location of cops) and \( r \) is a vertex in some component of \( G \setminus C \) (location of robber).

In Round 0, the cops choose \( C_0 \in \binom{V}{k} \) and then the robber chooses \( r_0 \in V \setminus C_0 \) arbitrarily.

In Round \( i, i > 0 \), the cops choose \( C_i \in \binom{V}{k} \) and then the robber chooses a vertex \( r_i \in V \setminus C_i \) such that there is a path between \( r_i \) and \( r_{i-1} \) in \( G \setminus (C_i \cap C_{i-1}) \).

The cops win if after some finite number of rounds the robber has no vertex to choose.
Cops, robber, and treewidth

**Theorem** [Seymour/Thomas ’93]. If a graph $G$ has treewidth at most $k$ then $k+1$ omniscient cops can catch a robber on $G$.

**Proof.** Suppose $n(G) > k+1$ and let $(\{X_i \mid i \in I\}, T = (I, F))$ be a smooth tree decomposition of width $\leq k$ for $G$.

Pick an arbitrary root $a \in I$ and for $i \in I$ let

$$V_i = \bigcup_{j \in T_i} X_j$$

where $T_i$ denotes the subtree rooted at $i$.

In the first round choose $C_0 = X_a$.

In Round $j$, we suppose $C_{j-1} = X_b$ for some $b \in I$ and $r_{j-1} \in V_b \setminus X_b$. Let $c$ be the child of $b$ for which $r_{j-1} \in V_c$. Observe that $X_b \cap X_c$ is a $k$-cut in $G$. Thus choosing $C_j = X_c$ confines the robber to $V_c \setminus X_c$.

After a finite number of steps the game will arrive at a leaf $\ell$ of $T$ for which $V_\ell \setminus X_\ell = \emptyset$ and the robber has nowhere to go. \(\square\)

**Remark.** The converse also holds but the proof is much more involved.
Cops and robber on the grid

**Proposition.** On the $n \times n$-grid $n - 1$ omniscient cops cannot catch a robber.

*Proof.* Whichever positions the $n - 1$ cops choose to occupy, there is always at least one row and at least one column of the grid without any cop.

We show that the robber can always move to the intersection of a cop-free row with a cop-free column.

Initially, this is clear.

Suppose that at some point one or more cops enter the free row and/or free column where the robber is located. Then the robber can move along the previously free row to the to-be free column and within this column to the to-be free row.

\[ \square \]

**Proposition.** On the $n \times n$-grid $n$ omniscient cops cannot catch a robber, for $n \geq 2$.

→ Exercise.

**Corollary.** The $n \times n$-grid has treewidth $n$.

**Corollary.** There are planar graphs on $n$ vertices whose treewidth is $\Omega(\sqrt{n})$. 
Partial k-trees

Definition. A $k$-tree is a graph formed from a $k$-clique by iteratively joining a new vertex to some $k$-clique.
In other words, a graph is a $k$-tree $\iff$ there is an order $\pi = (v_1, v_2, \ldots, v_n)$ of its vertices such that the neighbors of $v_i$ preceding it in $\pi$ form a $\min\{i-1, k\}$-clique, for all $1 \leq i \leq n$.

Observation.

a) 1-trees are exactly trees.

b) A $k$-tree on $n \geq k$ vertices has $kn - \binom{k+1}{2}$ edges.

Definition. A graph is a partial $k$-tree if it is a subgraph of a $k$-tree.
Partial k-trees and treewidth

**Theorem.** A graph $G$ is partial $k$-tree $\iff G$ has treewidth at most $k$.

**Proof.** "$\Leftarrow$": Let $({X_i \mid i \in I}, T = (I, F))$ be a smooth tree decomposition of width $\leq k$ for $G$. Induction on $|I|$.

$|I| = 1$: $G$ is subgraph of $K_{k+1}$, a $k$-tree.

Otherwise, let $i \in I$ be a leaf of $T$. Then there is a $v \in X_i$ that does not occur in any $X_j$, $j \in I \setminus \{i\}$. Removal of $i$ from $I$ results in a tree decomposition of width $\leq k$ for $G' = (V \setminus \{v\}, E \cup (X_i \setminus \{v\}))$.

By induction $G'$ is subgraph of some $k$-tree $H$. Add $v$ to $H$ and connect it to the $k$-clique $X_i \setminus \{v\}$. Clearly $G$ is a subgraph of the resulting $k$-tree. $\square$
Partial $k$-trees and treewidth

**Theorem.** A graph $G$ is partial $k$-tree $\iff G$ has treewidth at most $k$.

**Proof.** “$\Rightarrow$”: Let $H$ be a $k$-tree containing $G$ and $\pi = (v_1, \ldots, v_n)$ a vertex order for $H$ such that the neighbors of $v_i$ preceding it in $\pi$ form a min$\{i-1, k\}$-clique, for all $1 \leq i \leq n$.

Build a tree decomposition of width $k$ for $V_i = \{v_1, \ldots, v_i\}$ inductively such that for every $j$, $1 \leq j \leq i$, there is a node that contains $\{v_j\} \cup \Pi_j$, where $\Pi_j = V_j \cap N_H(v_j)$.

$i \leq k + 1$: A single node for $V_i$ suffices.

Otw, let $\ell = \max\{1 \leq \ell < i \mid v_\ell \in \Pi_j\}$. By the induction hypothesis there is a tree decomposition of width $k$ for $V_{i-1}$ in which one node $a$ contains $\{v_\ell\} \cup \Pi_\ell$.

Create a new node $b$, make it adjacent to $a$ only, and set $X_b = \{v_i\} \cup \Pi_i$. (Note that $\Pi_i \subseteq \{v_\ell\} \cup \Pi_\ell$ because $\Pi_i$ is a clique.) \qed
Grids, minors, and treewidth

**Theorem [Alon, Seymour, Thomas '90].** For any fixed graph $H$, every graph $G$ that does not contain $H$ as a minor has treewidth at most $n(H)^{3/2} \sqrt{n(G)}$.

**Corollary.** A planar graph on $n$ vertices has treewidth $O(\sqrt{n})$.

**Theorem [Robertson, Seymour, Thomas '94].** Every graph of treewidth larger than $20^{2k^5}$ has a $k \times k$-grid as a minor. There are graphs of treewidth $\Omega(k^2 \log k)$ that do not have a $k \times k$-grid as a minor.