# Independent Set

[Kleinberg and Tardos 2005
Chapter 10.2 and 10.4]

**Input**: Social Networks $(V, E)$

**Output**: $V' \subseteq V$

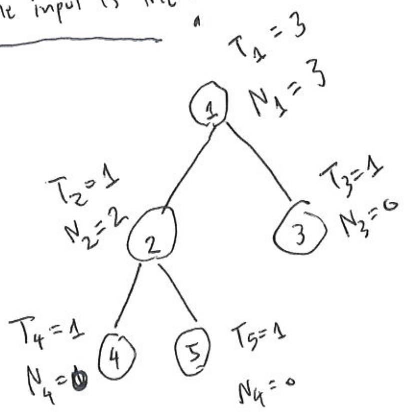
**Constraint**: For any $u, v \in V'$, $\{u, v\} \notin E$ [We don't want a friend to be in the party]

**Objective Function**: Maximize $|V'|$ [We want to maximize # persons who join the party]

NP-Hard = Unlikely to be solved.

When the input is tree?

Ex




Information if we take 2, 3 or not is the only information for 1.

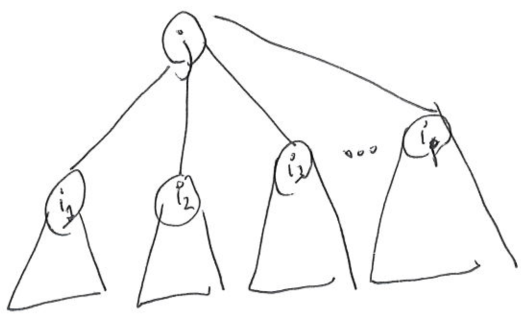
$T_i$ := max # nodes in $i$ and descendant when we include $i$

$N_i$ := max # nodes in $i$ and descendant when we do not include $i$.

From Bottom to Top:

$$T_j = \sum_{k=1}^{p} N_{i_k} + 1$$

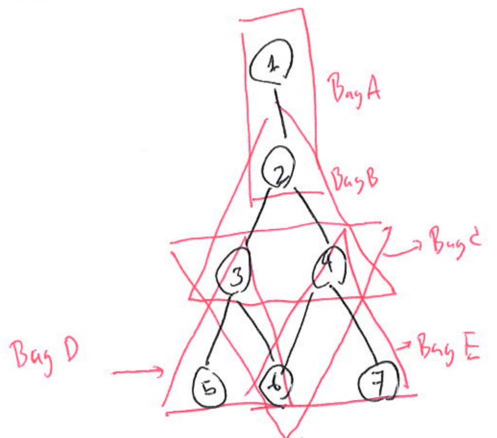
$$N_j = \sum_{k=1}^{p} \max\{T_{i_k}, N_{i_k}\}$$




Maximum # persons to invite $= \max\{T_{top}, N_{top}\}$

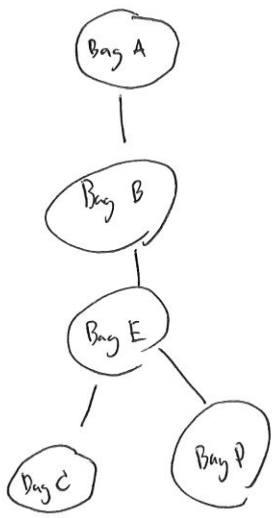
Hard problem is easy to solve when the input graph is a tree.

How about when the input graph is close to a tree?




almost tree!

We should have some fast algorithm for such a graph!




a tree of bags

Solving problems here!

# Tree Decomposition

**Input:** Social Network $(V, E)$

**Output:** Tree $(\beta, \Sigma)$

$\beta$: set of bags; $\Sigma$: links between bags

For each bag $\beta_i \in \beta$, a set of nodes associating to the bag $V_i \in V$

**Ex**

$V = \{1,2,3,4,5,6,7\}$

$E = \{\{1,2\},\{2,3\},\{2,4\},\{3,5\},\{3,6\},\{4,6\},\{4,7\}\}$

$\Downarrow$

$\beta = \{\beta_1, \dots, \beta_5\}$ $\quad \Sigma = \{\{\beta_1,\beta_2\},\{\beta_2,\beta_5\},\{\beta_3,\beta_5\},\{\beta_4,\beta_5\}\}$

$V_1 = \{1,2\}$ $\quad V_2 = \{2,3,4\}$, $V_3 = \{3,5,6\}$, $V_4 = \{4,6,7\}$, $V_5 = \{3,4,6\}$.

**Constraint**

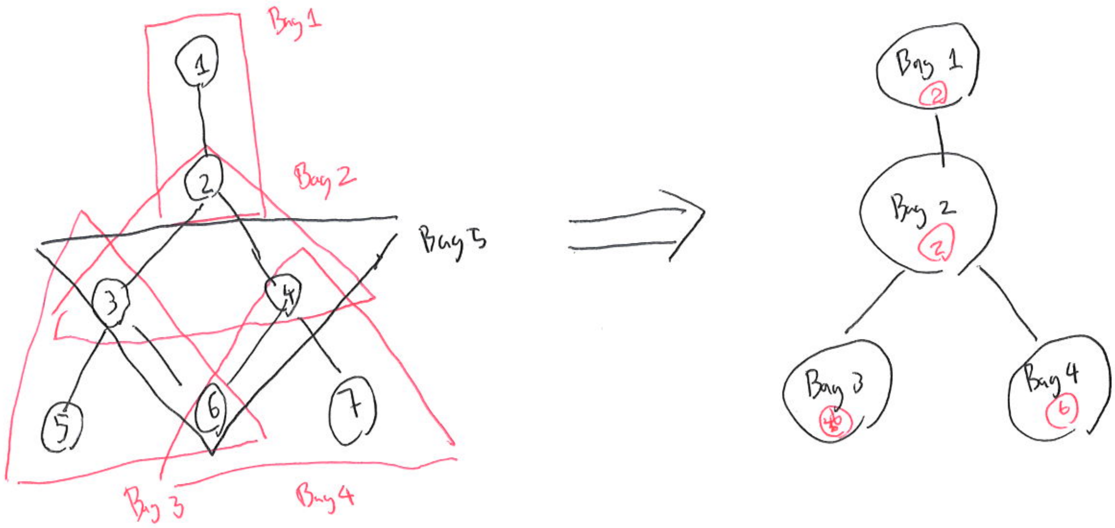
1. **Node Coverage** (all nodes must be in some bags)

$$\bigcup_i V_i = V$$

2. **Edge Coverage** (all edges must be in some bags)

Forall $\{u,v\} \in E$, there must be some bags $\beta_i$ such that $u,v \in V_i$

3. **Coherance** (Bags that have a specific node must connect to each other.)



Graph induced by Bags contain 2


also include edges that have 2 sides in considered node.

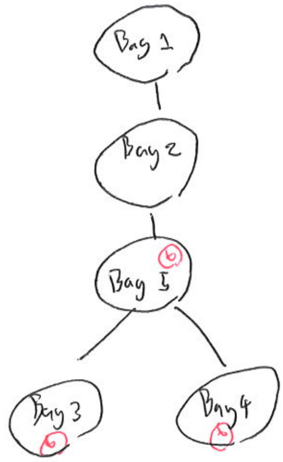
Bag 1 and Bag 2 can communicate each other

Graph induced by Bags contain 6

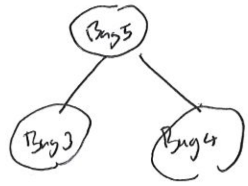

Bag 3 and Bag 4 cannot communicate.

This is not a tree decomposition!

With Bag 5!




Graph induced by Bags contain 6




Bags 3, 4, 5 can connect together by 2 edges.

Formal Definition

Assume that node $i$ contains in $V_{i_1}, V_{i_2}, \dots, V_{i_q}$

$$\beta^{(i)} := \{\beta_{i_1}, \beta_{i_2}, \dots, \beta_{i_q}\}$$

$$\Sigma^{(i)} := \{\{\beta_i, \beta_j\} \in \Sigma : \beta_i, \beta_j \in \beta^{(i)}\}$$

Induced subgraph by node $i$ : $(\beta^{(i)}, \Sigma^{(i)})$

Coherence : All induced subgraphs by node $i_q$ are connected.

Objective Function

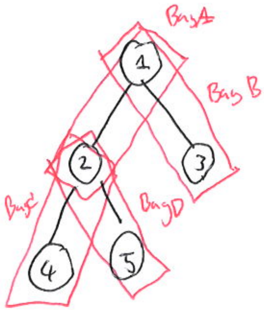
Minimize the size of largest bag

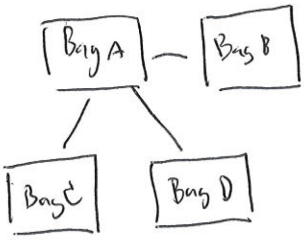
$$\text{Treewidth} = \max_i |V_i| - 1$$

Minimize Treewidth.

Ex Treewidth of our example = 3-1 = 2

Treewidth of a tree = 2-1 = 1




Tree width = 1 → tree

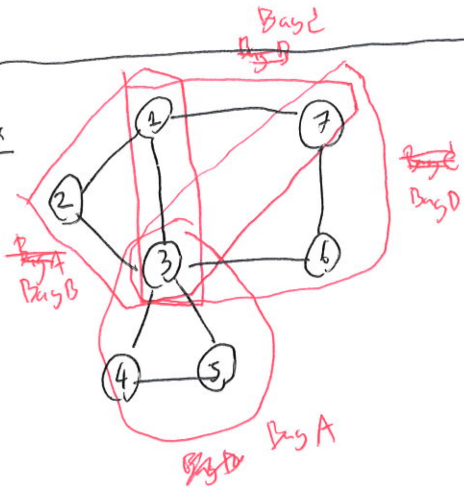
Tree width = 2 → close to tree

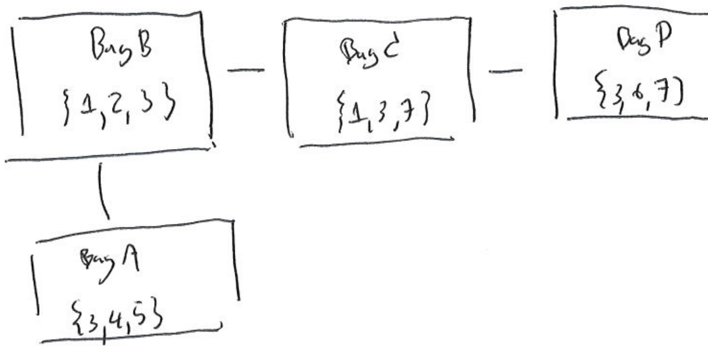
⋮

large treewidth → not close to tree

---

Ex

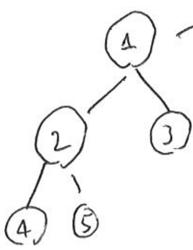



Treewidth = 3-1 = 2

---

Back to independent set




To calculate $T_1$ and $N_1$, we care if 2 and 3 are taken. We don't care if 4 or 5 are taken.

⇐ We don't have direct communication from 4 or 5 to 1