CS 5003: Parameterized Algorithms

Lectures 36-39

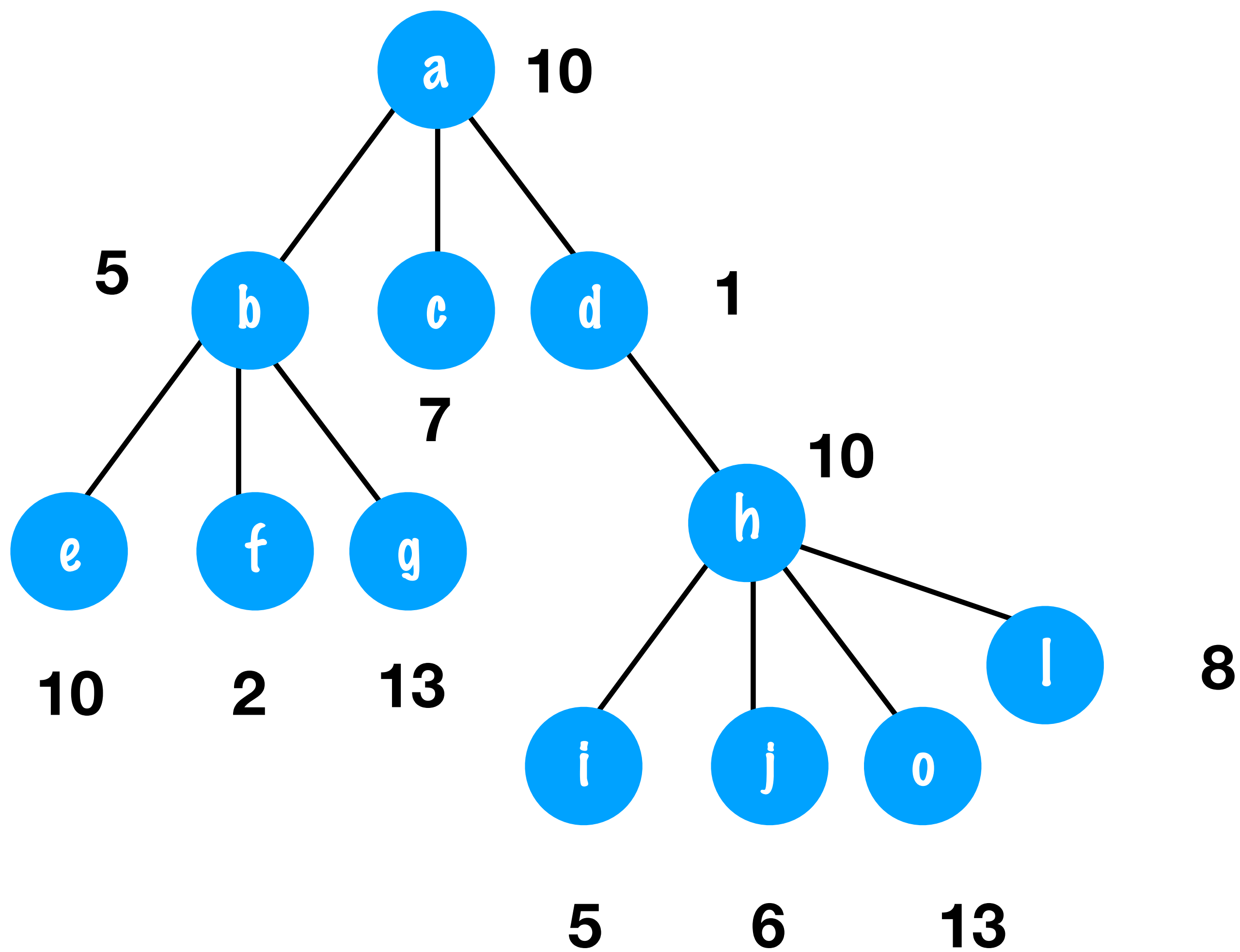
Krithika Ramaswamy

IIT Palakkad

Reference Books: Parameterized Algorithms by Cygan et al. and Kernelization by Fomin et al.

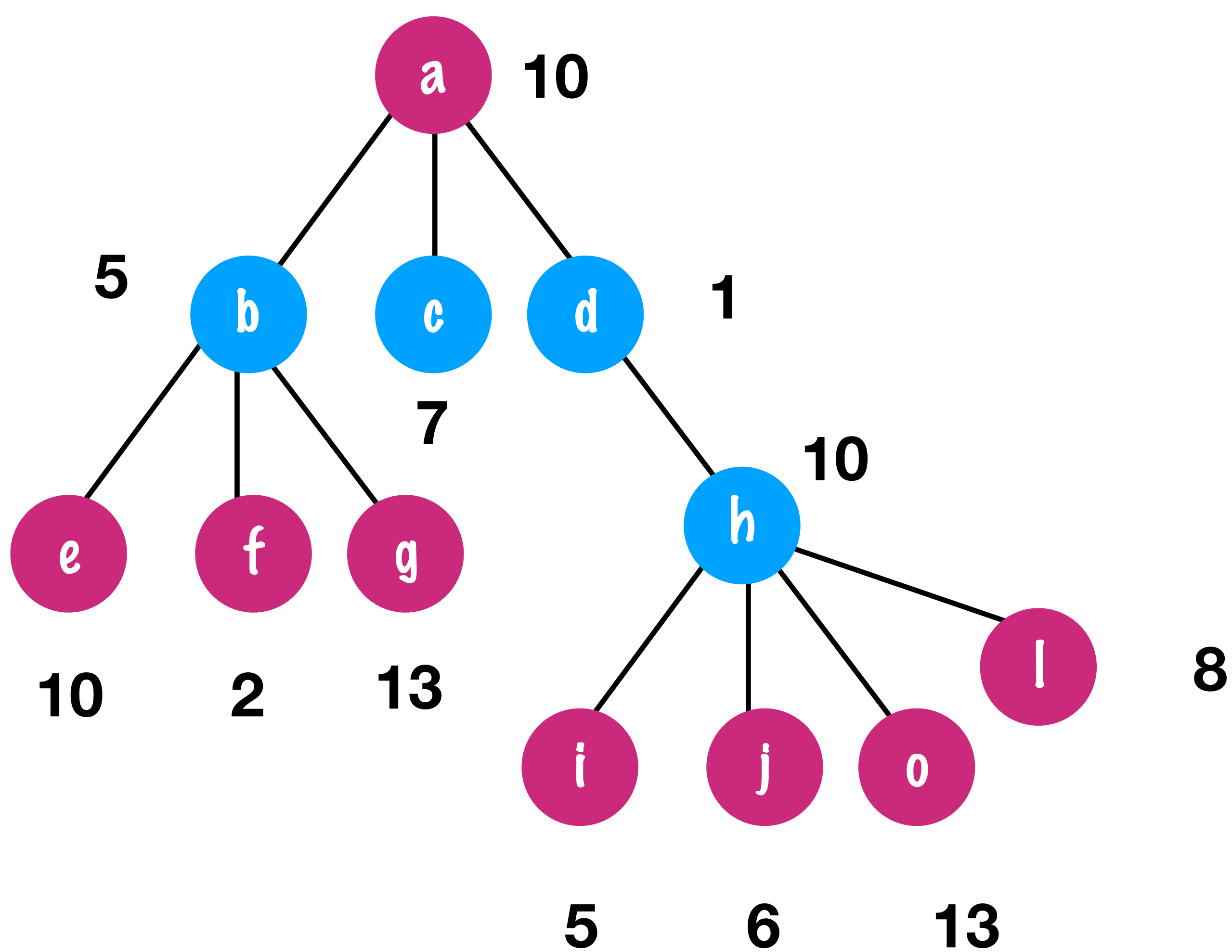
Independent Set on Trees

Weighted Independent Set
Instance: A tree T with positive integral weights on its vertices and an integer k
Question: Does there exist an independent set of T of weight at least k?



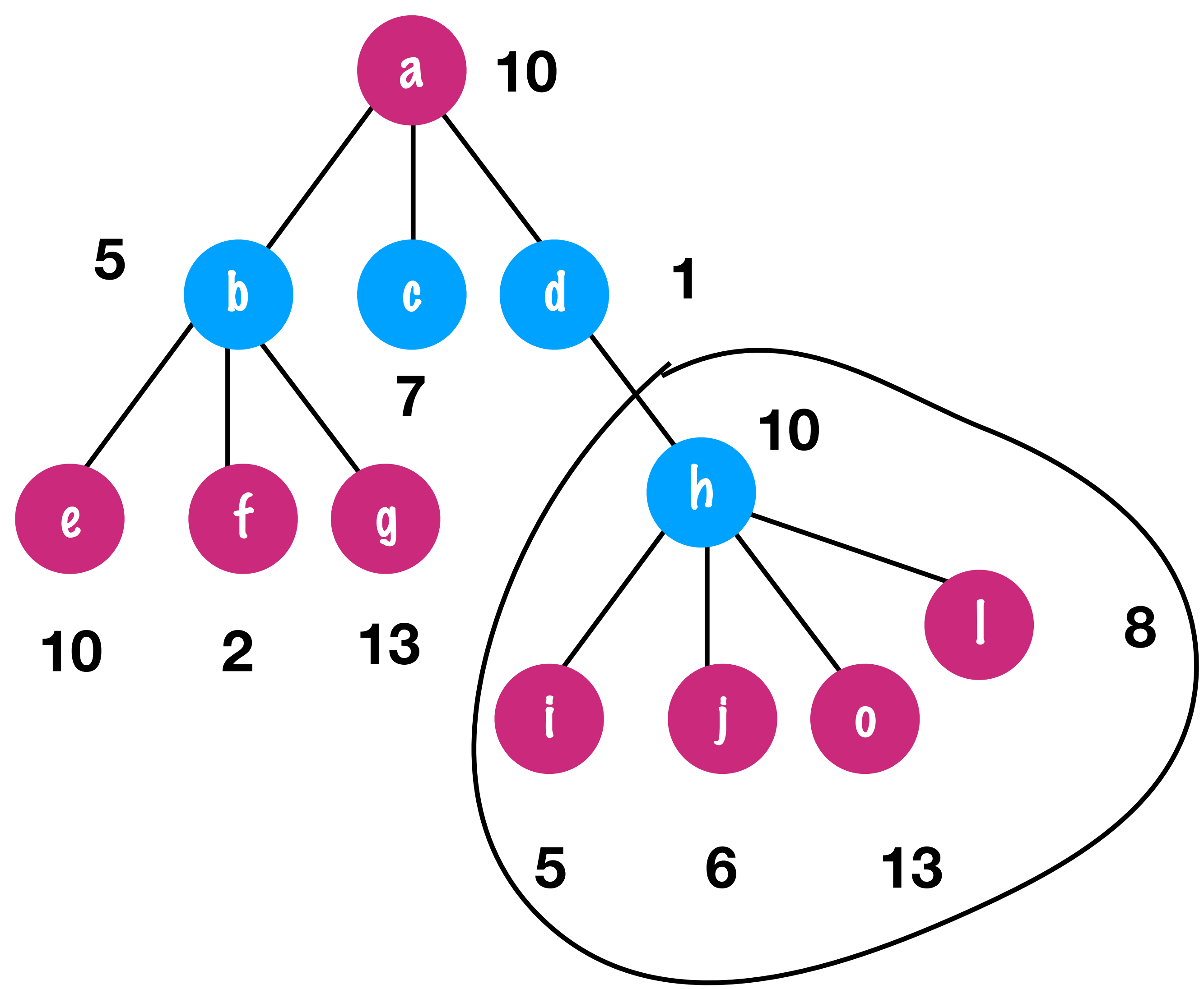
Independent Set on Trees

- Root T at an arbitrary vertex
- For a vertex v, let T_v denote the subtree of T rooted at v



Independent Set on Trees

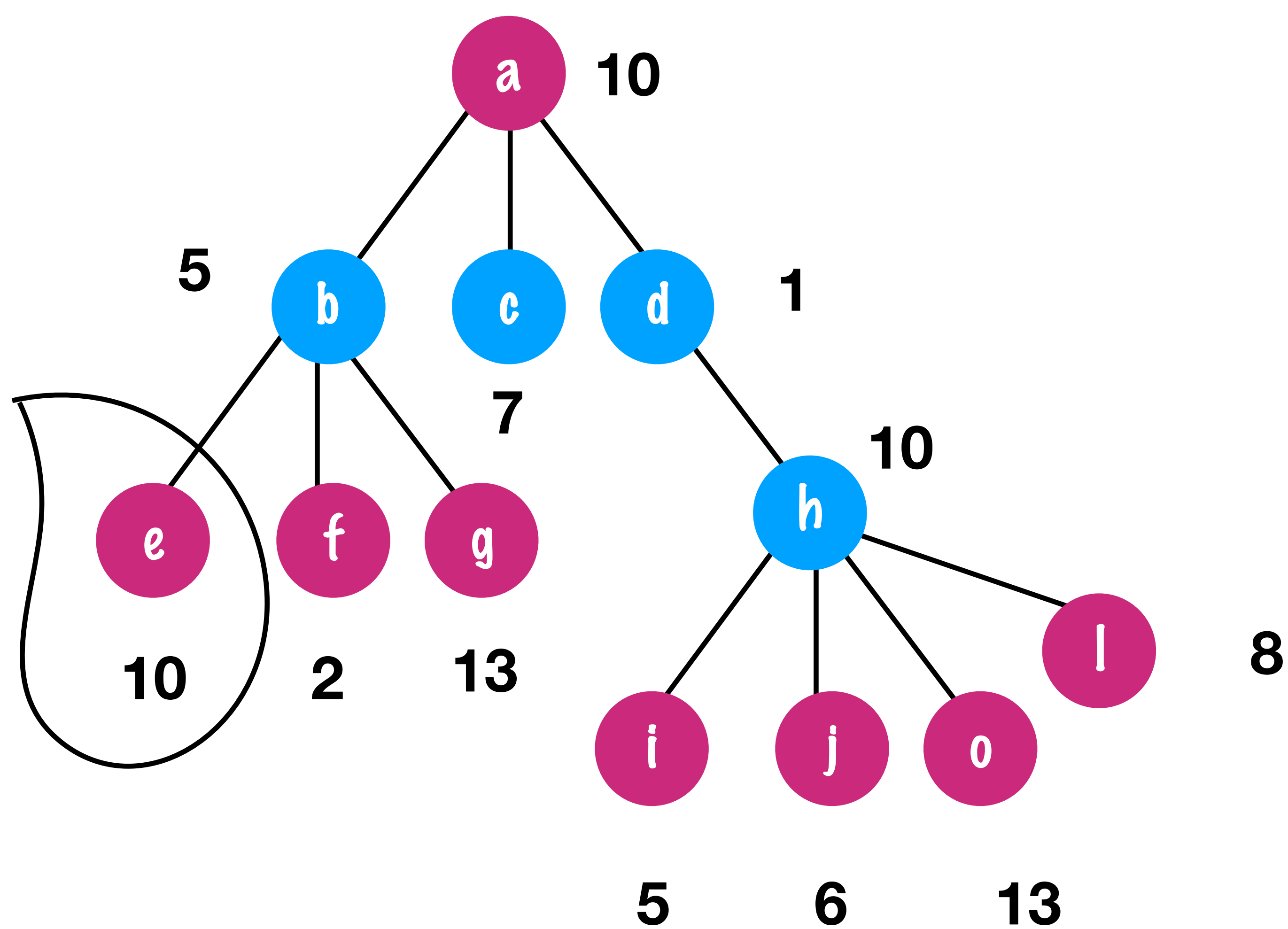
* Root T at an arbitrary vertex
* For a vertex v, let T_v denote the subtree of T rooted at v



* $\Lambda(h)$ = max possible wt of an IS in T_h that does not contain h

Independent Set on Trees

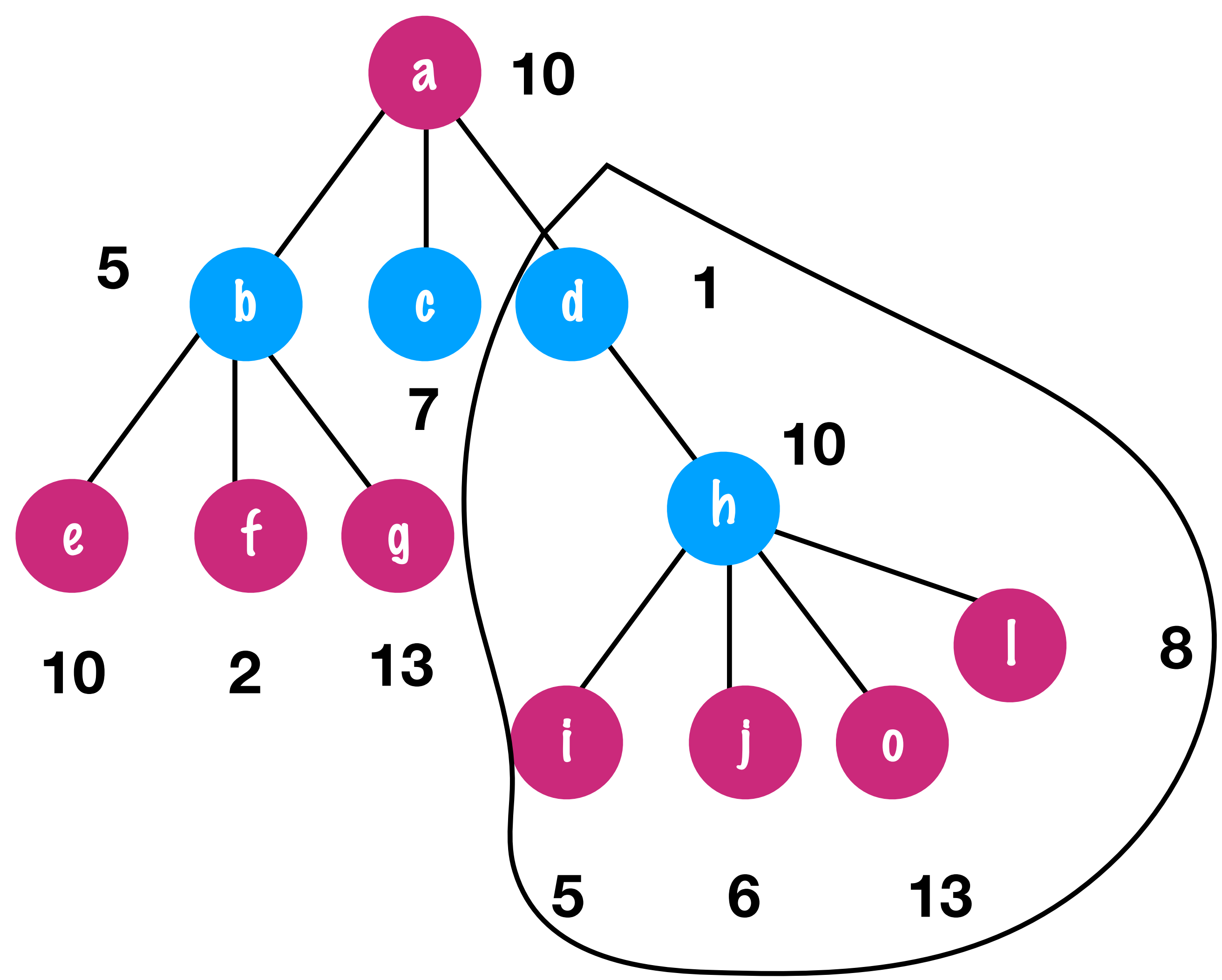
* Root T at an arbitrary vertex
* For a vertex v, let T_v denote the subtree of T rooted at v



* $\Gamma(e)$ = max possible wt of an IS in T_e

Independent Set on Trees

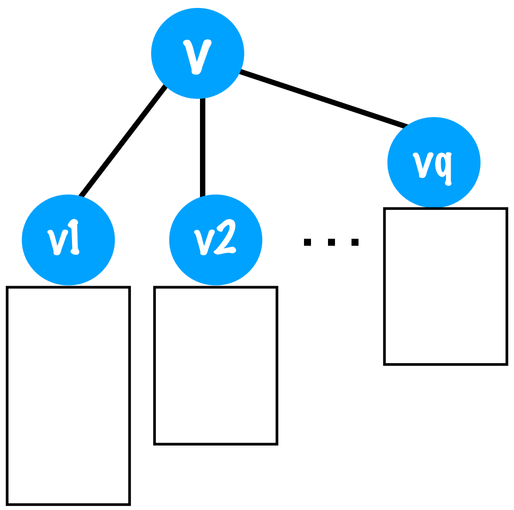
- Root T at an arbitrary vertex
- For a vertex v, let T_v denote the subtree of T rooted at v



- $\Gamma(d)$ = max possible wt of an IS not containing d in T_d

Independent Set on Trees

* Suppose v has $v_1, v_2, \ldots, v_q$ as its children

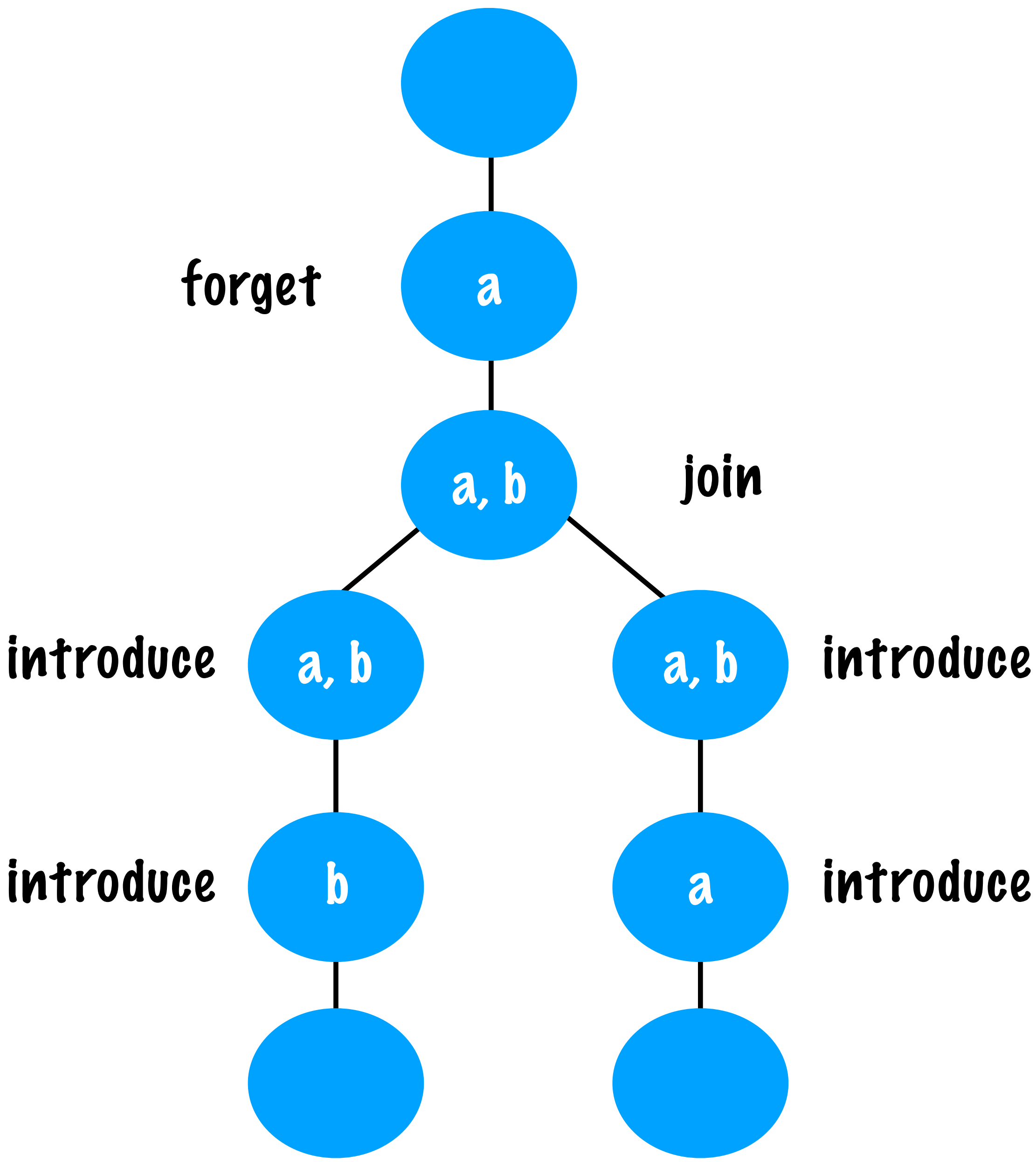


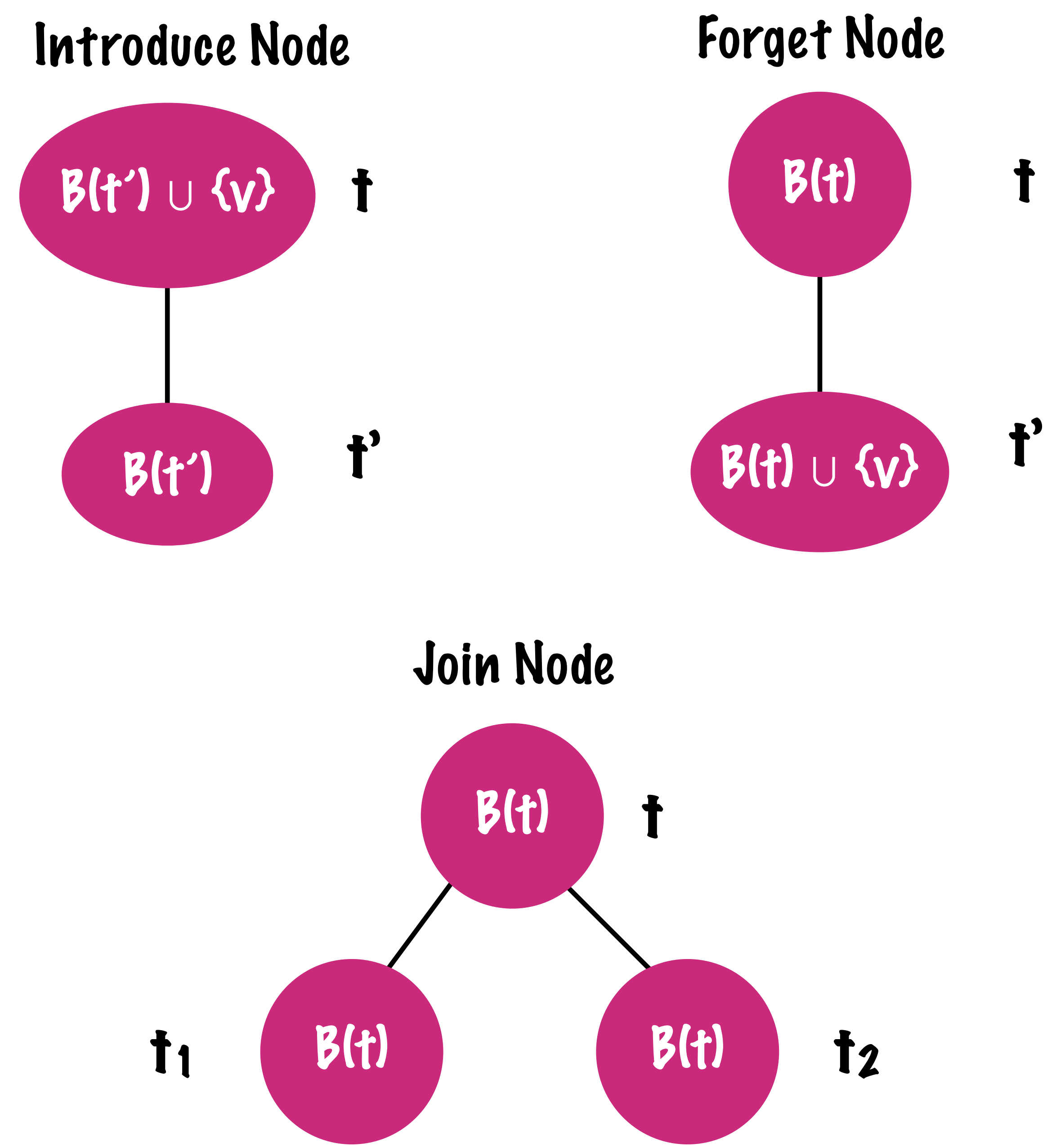
* $\Gamma(v)$ = max possible wt of an IS in T_v
* $\Lambda(v)$ = max possible wt of an IS in T_v that does not contain v
  * $\Lambda(v) = \Gamma(v_1) + \ldots + \Gamma(v_q)\}$
  * $\Gamma(v) = \max\{\Lambda(v), w(v) + \Lambda(v_1) + \ldots + \Lambda(v_q)\}$
* Computing $\Lambda(v)$ and $\Gamma(v)$ for leaves is easy

Linear time algorithm

Independent Set on Nice Tree Decompositions

Weighted Independent Set
Instance: A graph G with positive integral weighting on its vertices, a nice tree decomposition (T, B) of G and an integer k
Question: Does there exist an independent set of G of weight at least k?
Parameter: w(T)



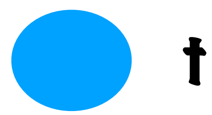


Independent Set on Nice Tree Decompositions

- For a node t in T
  - Let V_t be the union of all bags in the subtree of T rooted at t
- For every t in T and every $S \subseteq B(t)$
  - Let $\Gamma(t, S)$ denote the max possible wt of an IS S^* s.t
    - $S \subseteq S^* \subseteq V_t$
    - $S^* \cap B(t) = S$
    - If S^* does not exist then $\Gamma(t, S) = -\infty$

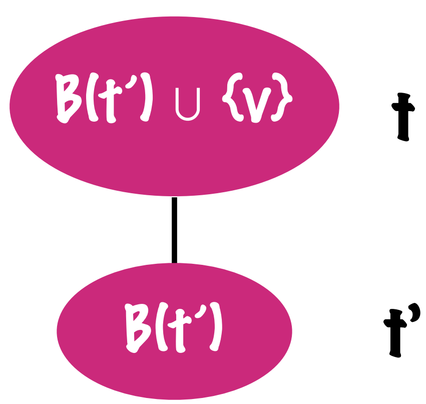
Independent Set on Nice Tree Decompositions

Leaf node:



$\Gamma(t, \emptyset) = 0$

Introduce node:

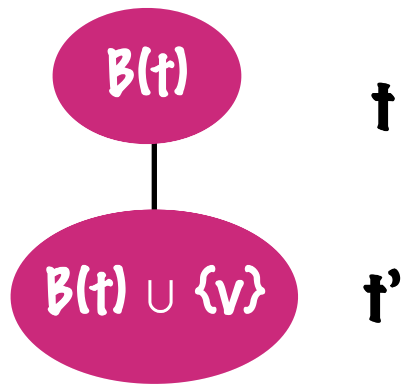


$\Gamma(t, S) = \Gamma(t', S)$ if v is not in S

$\Gamma(t, S) = w(v) + \Gamma(t', S \setminus \{v\})$ if v is in S

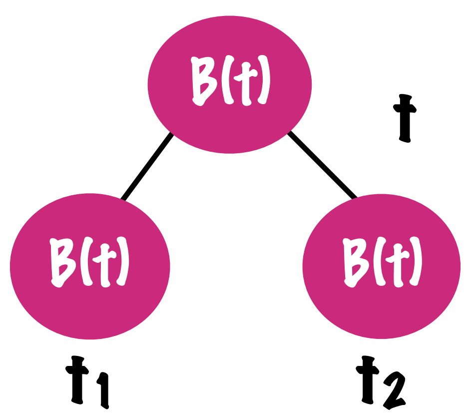
Independent Set on Nice Tree Decompositions

Forget node:



$$\Gamma(t, S) = \max\{\, \Gamma(t', S),\ \Gamma(t', S \cup \{v\}) \,\}$$

Join node:



$$\Gamma(t, S) = \Gamma(t_1, S) + \Gamma(t_2, S) - w(S)$$

Independent Set on Nice Tree Decompositions

Analysis:

* For any node t in T, $|B(t)| <= w(T) + 1$
* At node t, we compute $2^{|B(t)|} <= 2^{(w(T) + 1)}$ values of $\Gamma(t, .)$
    * For a fixed S, computing $\Gamma(t, S)$ is polynomial time
* No. of nodes in T is $O(w(T)*n)$
* $\Gamma(root, \varnothing)$ is the required answer

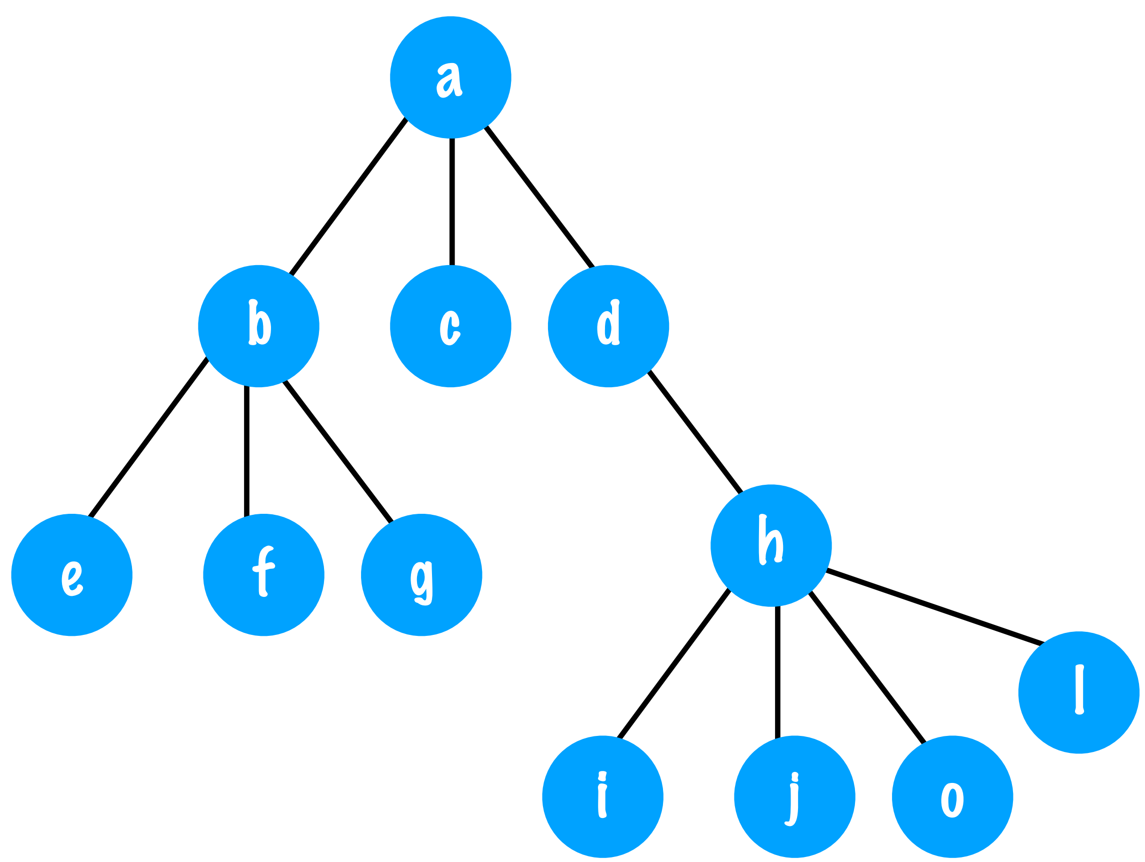
$2^{w(t)} n^{O(1)}$ time algorithm

Theorem: Weighted Independent Set parameterized by the treewidth of the input graph is FPT.

Dominating Set on Trees

Definition: A dominating set in G is a set S of vertices such that $N[S] = V(G)$

Dominating Set
<u>Instance:</u> A tree T and an integer k
<u>Question:</u> Does there exist a dominating set of T of size at most k?



Dominating Set on Trees

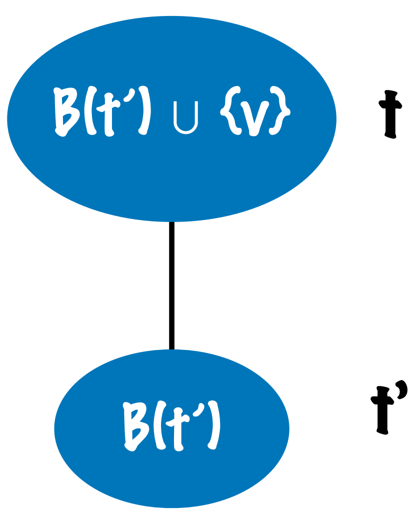
* Root T at an arbitrary vertex
* For a vertex v, let T_v denote the subtree of T rooted at v
* Suppose v has $v_1, v_2, \ldots, v_q$ as its children
  * Let $\Gamma(v)$ denote the min possible size of a Dom Set in T_v
  * Let $\Lambda(v)$ denote the min possible size of a set in T_v that dominates every vertex in T_v-v
  * Let $\Delta(v)$ denote the min possible size of a Dom Set in T_v that contains v
    * $\Delta(v) = 1 + \Lambda(v_1) + \ldots + \Lambda(v_q)$
  * $\Lambda(v) = \min\{\Gamma(v_1) + \ldots + \Gamma(v_q),\ 1 + \Lambda(v_1) + \ldots + \Lambda(v_q)\}$
  * $\Gamma(v) = \min\{1 + \Lambda(v_1) + \ldots + \Lambda(v_q),\ \min\{\Delta(v_i) + \Sigma_{i \neq j}\, \Gamma(v_j) : i \in [q]\}\}$
* Computing $\Lambda(.)$, $\Delta(.)$ and $\Gamma(.)$ for leaves is easy

Linear time algorithm

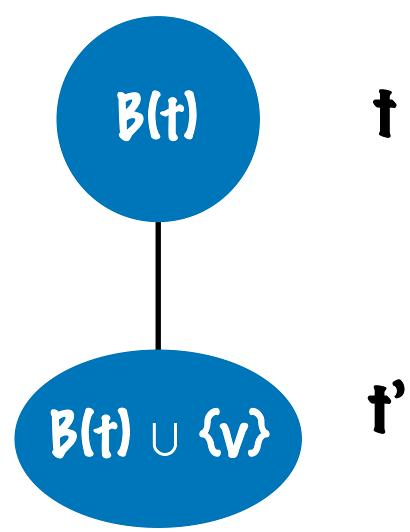
Dominating Set on Nicer Tree Decompositions

Dominating Set
Instance: A graph G, a nicer tree decomposition (T, B) of G and an integer k
Question: Does there exist a dominating set of T of size at most k?
Parameter: w(T)

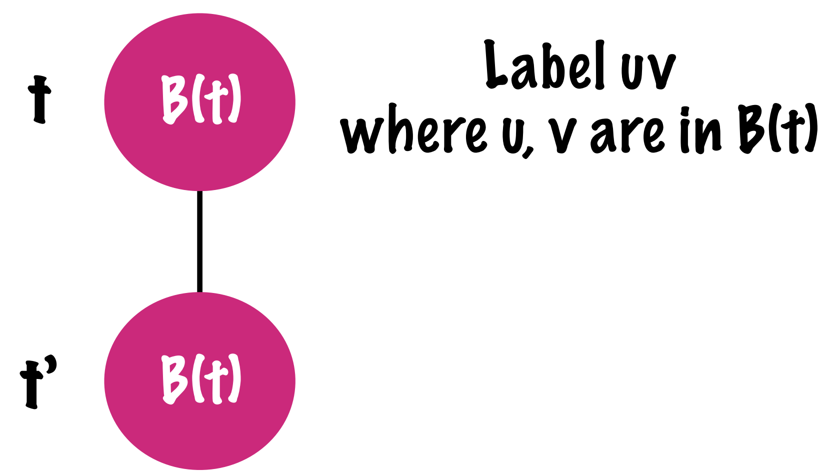
Introduce Node



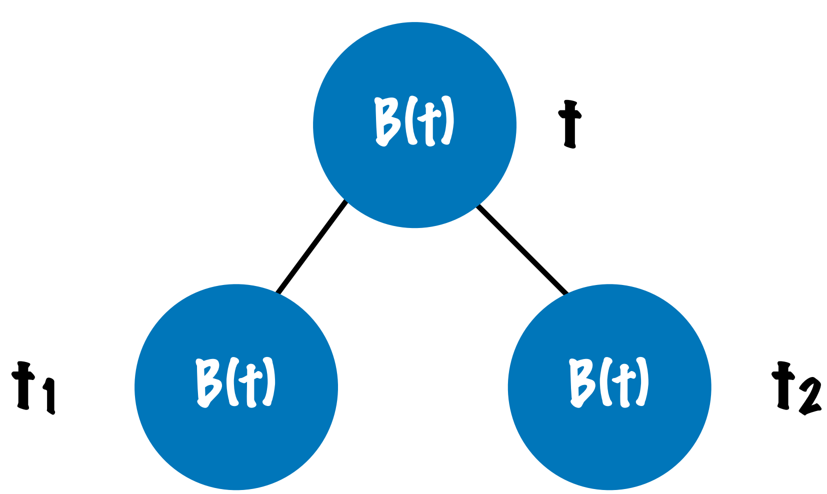
Forget Node



Introduce Edge Node



Join Node



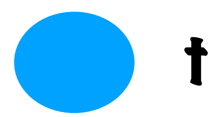
* For every edge {u, v} in G, there is exactly one introduce edge node with label uv

Dominating Set on Nicer Tree Decompositions

* For a node t in T,
  * Let V_t be the union of all bags in the subtree of T rooted at t
  * Let E_t be the edges in $G[V_t]$ introduced in the subtree of T rooted at t
  * Let G_t denote the subgraph of G with vertex set V_t and edge set E_t
* For a node t in T and a partition of B(t) into 3 sets X, Y and Z
  * Let $\Gamma(t, X, Y, Z)$ denote the min possible size of a set S^* in G_t s.t
    * $X \subseteq S^*$
    * S^* dominates every vertex in $V_t \setminus Z$
    * $Y \cap S^* = \varnothing$ and $Z \cap S^* = \varnothing$

Dominating Set on Nicer Tree Decompositions

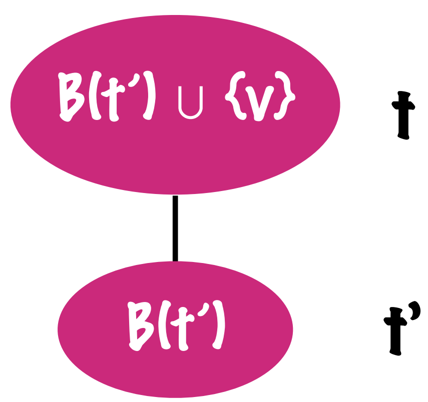
Leaf node:



$\Gamma(t, \emptyset, \emptyset, \emptyset) = 0$

polynomial time

Introduce vertex node:



$V_{t'} = V_t \setminus \{v\}$ and $E_{t'} = E_t$

v is isolated in G_t

$\Gamma(t, X, Y, Z) = \infty$ if v is in Y

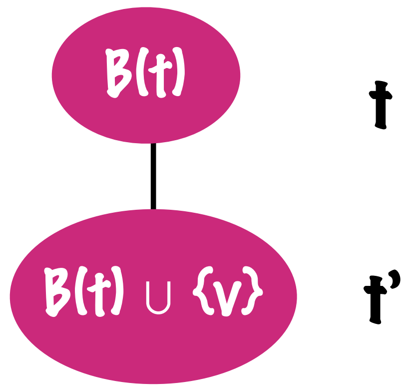
$\Gamma(t, X, Y, Z) = \Gamma(t', X, Y, Z \setminus \{v\})$ if v is in Z

polynomial time

$\Gamma(t, X, Y, Z) = 1 + \Gamma(t', X \setminus \{v\}, Y, Z)$ if v is in X

Dominating Set on Nicer Tree Decompositions

Forget node:



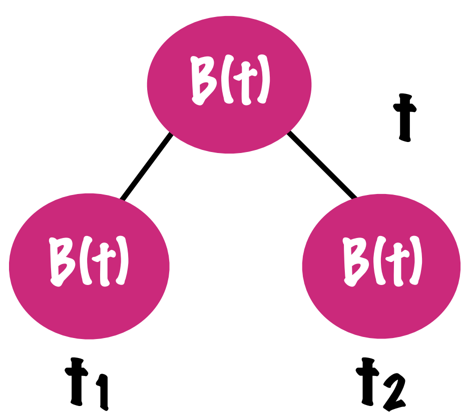
v has to be dominated!

Y = Y' \ {v}

$\Gamma(t, X, Y, Z) = \min \{ \Gamma(t', X', Y', Z') : v \notin Z', X = X' \setminus \{v\}, Y \subseteq Y' \}$

polynomial time

Join node:



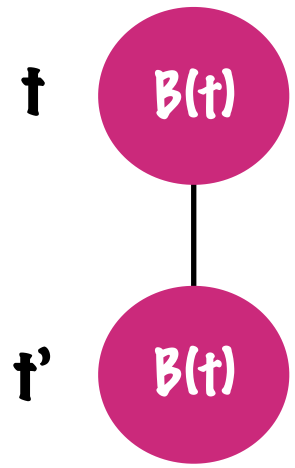
$2^{w(t)} n^{O(1)}$ time

Actually, Y = Y1 U Y2Zi = B(t) \ (X U Yi)

$\Gamma(t, X, Y, Z) = \min \{ \Gamma(t_1, X, Y_1, Z_1) + \Gamma(t_2, X, Y_2, Z_2) - |X| : Y \subseteq Y_1 \cup Y_2 \}$

Dominating Set on Nicer Tree Decompositions

Introduce edge node:



Label uv
where u, v are in B(t)

$V_{t'} = V_t$ and $E_t = E_{t'} \cup \{\{u,v\}\}$

u \in X => v is already dominated

If $v \in Y$ and $u \in X$ then,

$$\Gamma(t, X, Y, Z) = \min \{ \Gamma(t', X, Y, Z), \min \{ \Gamma(t', X, Y', Z') : Y \setminus \{v\} \subseteq Y', Z \subseteq Z' \}\}$$

If $v \in X$ and $u \in Y$ then,

$$\Gamma(t, X, Y, Z) = \min \{ \Gamma(t', X, Y, Z), \min \{ \Gamma(t', X, Y', Z') : Y \setminus \{u\} \subseteq Y', Z \subseteq Z' \}\}$$

Otherwise,

$$\Gamma(t, X, Y, Z) = \Gamma(t', X, Y, Z) \text{ if } v \in X \text{ and } u \in X \text{ or } v \notin X \text{ and } u \notin X$$

polynomial time

Dominating Set on Nicer Tree Decompositions

Analysis:

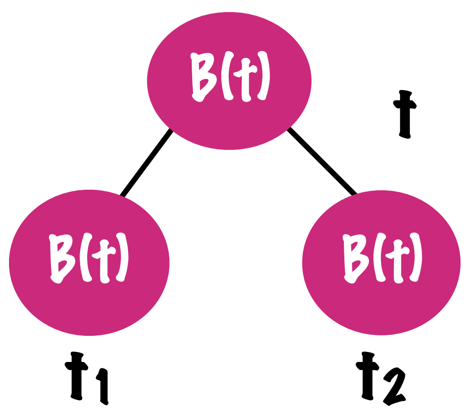
* For any node t in T, $|B(t)| \leq w(T) + 1$
* At node t, we compute $3^{|B(t)|} \leq 3^{(w(T)+1)}$ values of $\Gamma(t, ., ., .)$
  * For a fixed (X, Y, Z), compute $\Gamma(t, X, Y, Z)$ in
    * $2^{w(t)}$ time if t is a join node
    * Polynomial time if t is not a join node

$3^{w(t)}\ 2^{w(t)}\ n^{O(1)}$ time algorithm

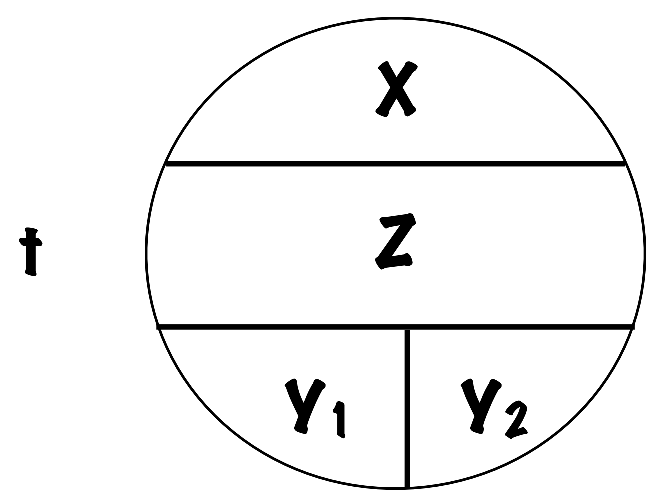
Theorem: Dominating Set parameterized by the treewidth of the input graph is FPT.

Dominating Set on Nicer Tree Decompositions

Join node:



$$\Gamma(t, X, Y, Z) = \min \{ \Gamma(t_1, X, Y_1, Z_1) + \Gamma(t_2, X, Y_2, Z_2) - |X| : Y \subseteq Y_1 \cup Y_2 \}$$



The min of $4^{w(t)}$ values is $\Gamma(t, X, Y, Z)$

Dominating Set on Nicer Tree Decompositions

Analysis:

* For any node t in T, $|B(t)| <= w(T) + 1$
* At a non-join node t, we compute $3^{|B(t)|} <= 3^{(w(T) + 1)}$ values of $\Gamma(t, ., ., .)$
    * For a fixed (X, Y, Z), compute $\Gamma(t, X, Y, Z)$ in polynomial time
* For a join node t, $4^{w(t)}$ time to compute $\Gamma(t, ., ., .)$

$4^{w(t)} n^{O(1)}$ time algorithm

Theorem: Dominating Set parameterized by the treewidth of the input graph is FPT.