Kernelization : Reducing a problem to itself

High level idea : Reduce a large instance to an "equivalent" smaller or simpler instance

Today: Given (G, k) an instance of Vertex Cover

Produce (G', k') another instance of Vertex Cover such that

- $k' \leq k$
- $|G'| = O(k^2)$
- (G', k') is a Yes instance $\iff$ (G, k) is a Yes instance

This can be done in polynomial time

<u>Main Obs</u>: If (G, k) is a Yes instance then

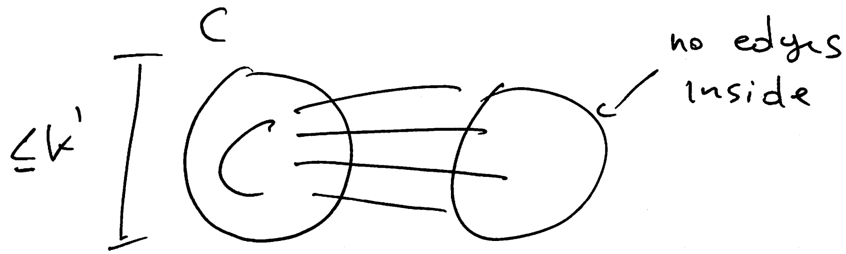
if $\exists u : \deg(u) > k$ then u <u>must</u> belong to minimum vertex cover

Rule 1 : If (G,k) has a vertex u : $\deg(u) > k$ then reduce to $(G-u, k-1)$

Rule 2 : If (G,k) has a vertex u : $\deg(u) = 0$ then reduce to $(G-u, k)$

Obs : If we repeatedly apply Rules 1 & 2 to (G,k) until it is not possible anymore and let (G',k') be the resulting instance, then

- $|V'| + |E'| = O(k'^2)$ if (G',k) is a Yes-instance
- (G,k) is a Yes-instance $\iff$ (G',k') is a Yes-instance



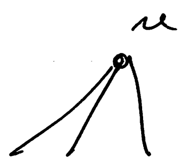
$$|E'| \leq \sum_{u \in C} \deg_{G'}(u) \leq \sum_{u \in C} k' \leq k'^2$$

$$|V'| \leq 2|E'|$$ ← since degrees are non-zero

Other kernelization Rules

$(G,k) \to (G',k')$ such that G' has min degree ≥ 2

Consider the branching rule:

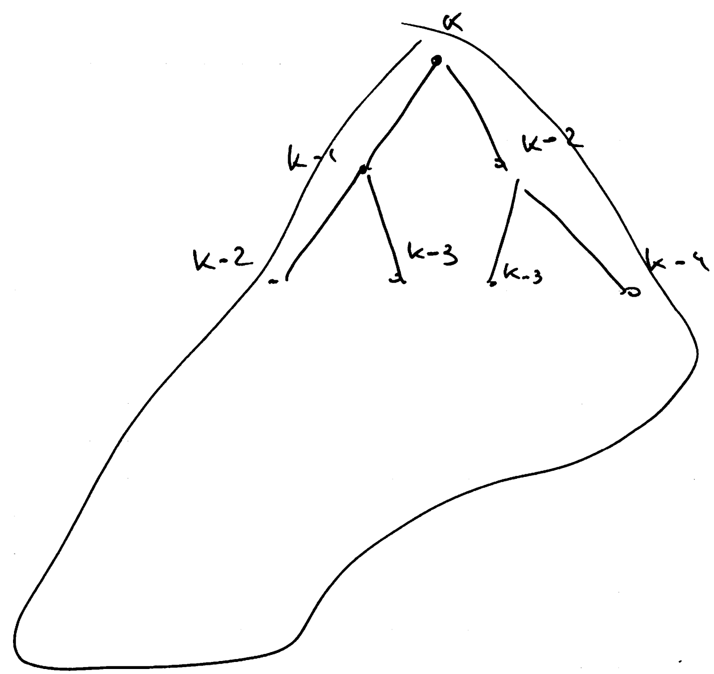


$u \in VC \Rightarrow$ add u to VC solve $(G-u, k-1)$

$u \notin VC \Rightarrow$ add $N(u)$ to VC solve $(G \setminus N(u), \underbrace{k-|N(u)|}_{\leq k-2})$

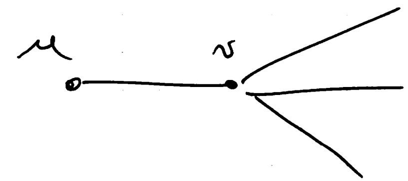
$$T(k) \leq T(k-1) + T(k-2) + \text{poly}(n)$$

$$\Rightarrow T(k) = O\left(1.62^k \text{ poly}(n)\right)$$

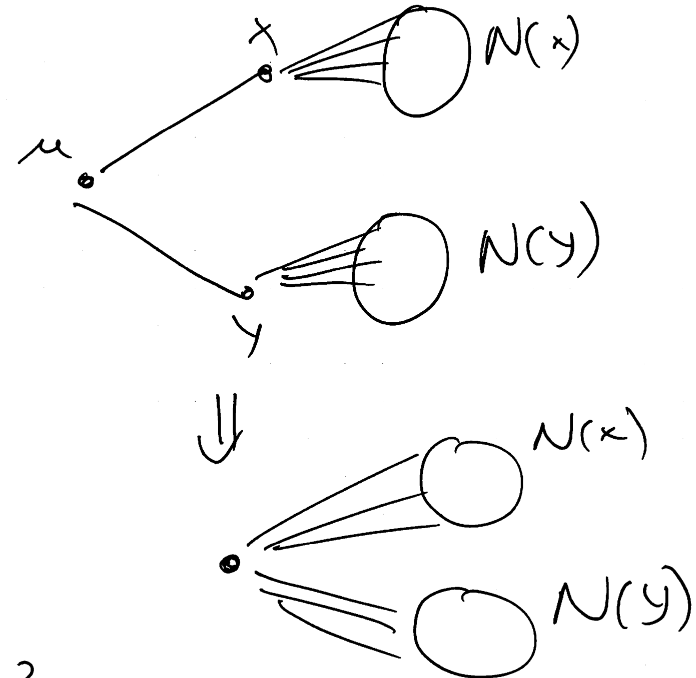


Rule 3 : If $\exists\, u : \deg(u) = 1$

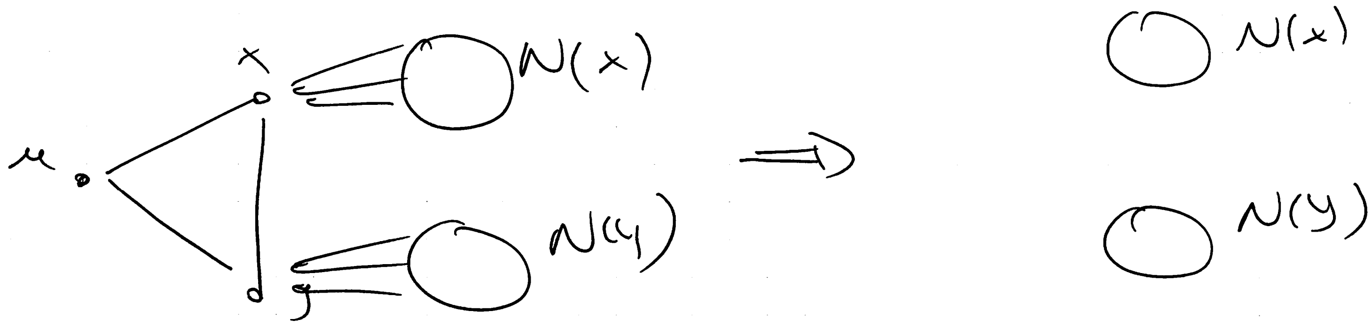
$$(G, k) \rightarrow (G \setminus \{u, v\}, k-1)$$



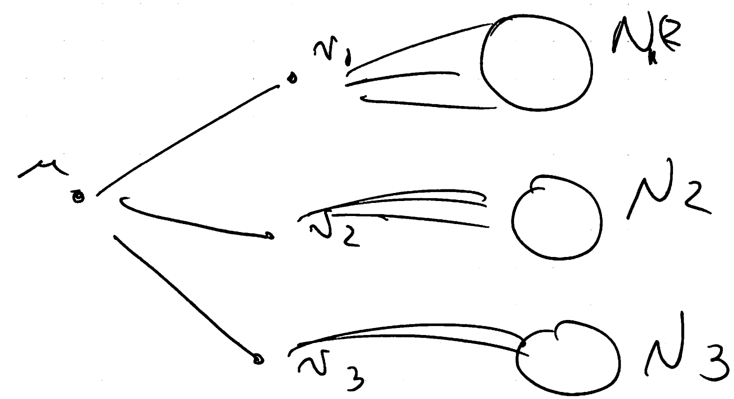
Rule 4 : If $\exists\, u : \deg(u) = 2$ and neighbors are not connected, then merge $\{u, x, y\}$ into a new node and set $k' = k - 1$



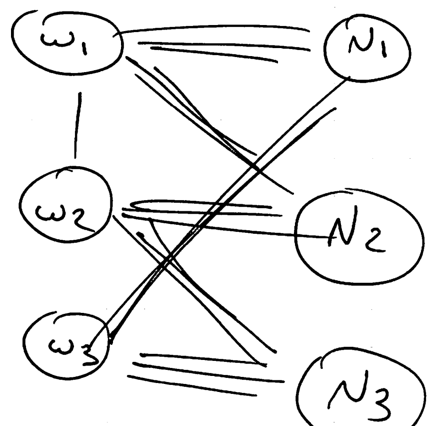
If $\exists\, u : \deg(u) = 2$ and neighbors are connected then remove $\{u, x, y\}$ and set $k' = k - 2$



Rule 5: If $\exists u$ deg(u) such that



replace with



(G, k) is Yes $\Leftrightarrow$ $(G', k$ [illegible]$)$ is Yes

All N_1 chosen?	All N_2 chosen?	All N_3 chosen?	How many from $\{u, v_1, v_2, v_3\}$ are needed?	How many from $\{w_1, w_2, w_3\}$ are?
F	F	F	3	3
F	F	T	3	3
F	T	F	3	3
F	T	T	2	2
⋮				