More definitions

- parameterized decision problem
- fixed-parameter tractable
- FPT

Example of a Parameterized Decision Problem: k-Vertex Cover

- Input: An undirected graph $G = (V,E)$, a positive integer k
- Parameter: k
- Question: Does there exist a vertex cover $V' \subseteq V$ for G that is of size at most k, that is $|V'| \leq k$?

Parameterized Decision Problems

- A parameterized decision problem is a decision problem with additionally an identified parameter.
- A parameter can be a part of the problem's input, a property of the problem's input or a combination of several parameters
- The running time analysis of an algorithm solving a parameterized decision problem is done in terms of the input size, n, and its parameter, k

Fixed parameter tractable parameterized decision problems & the parameterized complexity class FPT

A parameterized decision problem with input size n and parameter k that can be solved by an algorithm in time $O(f(k)\ n^c)$, where $c > 0$ is a constant, is called *fixed-parameter tractable,* and is a member of the class *FPT,* the class of fixed-parameter tractable parameterized decision problems.

f denotes any (computable) function. That is, f may be exponential or worse, but depends only on k.

Examples: $f(k) = 2^k$, $f(k) = k^k$, $f(k) = k^{128k}$

Observation

A parameterized decision problem with input size n and parameter k that can be solved by an algorithm in time $O(f(k) + n^c)$, where $c > 0$ is a constant, is fixed-parameter tractable.

Solving Vertex Cover: Polynomial-time Preprocessing

- Observation: Given are graph $G = (V,E)$ and integer $k > 0$. If G has a vertex cover V' of size at most k, and if there exists a vertex $v \in V$ with $\deg(v) = 0$, then $v \notin V'$.

- This results in the following preprocessing step for solving k-Vertex Cover.

- Rule 1. *Clean-up singletons:*
  while there exists $v \in V$ with $\deg(v) = 0$ **do**
  $$G \leftarrow G\text{-}v$$

Solving Vertex Cover: Polynomial-time Preprocessing

- Observation: Given are graph $G = (V,E)$ and integer $k > 0$. If G has a vertex cover V' of size at most k, and if there exists a vertex $v \in V$ with $\deg(v) > k$, then $v \in V'$.

- This results in the following preprocessing step for solving k-Vertex Cover.

- Rule 2. *Pick high-degree vertices:*
  while there exists $v \in V$ with $\deg(v) > k$ **do**
  $V' \leftarrow V' \cup \{v\}$
  $k \leftarrow k\text{-}1$
  $G \leftarrow G\text{-}v$

Solving Vertex Cover: Polynomial-time Preprocessing

- Note that if Rule 2 cannot be applied, then every vertex has degree at most k. Further:

- Given are graph $G = (V,E)$ and integer $k > 0$ where Rule 2 cannot be applied. If G has a vertex cover of size at most k, then $|V| \leq k^2+k$.

Solving Vertex Cover: Polynomial-time Preprocessing

- Observation: Given are graph $G = (V,E)$ and integer $k > 0$. If there exists a vertex $v \in V$ with $\deg(v) = 1$, then v's neighbour w can be included in any vertex cover, that is $w \in V'$.

- This results in the following preprocessing step for solving k-Vertex Cover.

- Rule 3. *Pick neighbours of pendant vertices:*
  while there exists $v \in V$ with $\deg(v) = 1$ **do**
  $V' \leftarrow V' \cup \mathrm{N}(v)$
  $k \leftarrow k\text{-}1$
  $G \leftarrow G\text{-}v\text{-}\mathrm{N}(v)$

Solving Vertex Cover: Polynomial-time Preprocessing

- Observation: Given are graph $G = (V,E)$ and integer $k > 0$. If there exists a vertex $v \in V$ with $\deg(v) = 2$, $\mathrm{N}(v) = \{w,z\}$, and edge $(w,z) \in E$ then v's neighbours w and z can be included in any vertex cover, that is $w,z \in V'$.

- This results in the following preprocessing step for solving k-Vertex Cover.

- Rule 4. *Pick neighbours of pendant vertices:*
  while there exists $v \in V$ with $\deg(v) = 2$, $\mathrm{N}(v) = \{w,z\}$, and edge (w,z) $\in E$ **do**

  $V' \leftarrow V' \cup \mathrm{N}(v)$

  $k \leftarrow k\text{-}2$

  $G \leftarrow G\text{-}v\text{-}\mathrm{N}(v)$

Solving Vertex Cover: Polynomial-time Preprocessing

- We summarize the polynomial-time preprocessing as follows.

- **repeat until no change occurs:**
    - **while** Rule 1 applies **do** apply Rule 1
    - **while** Rule 2 applies **do** apply Rule 2
    - **while** Rule 3 applies **do** apply Rule 3
    - **while** Rule 4 applies **do** apply Rule 4

Solving Vertex Cover: Polynomial-time Preprocessing

- Further, once none of the preprocessing rule applies we can make the following decision, since if the graph has a vertex cover of size at most k then $|V| \leq k^2+k$:

- **if** $|V| > k^2+k$ **then** answer “no”

Solving Vertex Cover: Polynomial-time Preprocessing

- After the preprocessing steps are done, and the algorithm didn’t return “no”, the bounded search tree algorithm can be run next to decide the problem.

- This results in a running time of $O(kn + 2^k k^2)$.