1 Introduction

1.1 Vertex Cover

A vertex cover in a graph $G = (V, E)$ is a subset of its vertices $S \subseteq V$ such that every edge of $G$ has at least one endpoint in $S$.

**Vertex Cover**

<table>
<thead>
<tr>
<th>Input:</th>
<th>A graph $G = (V, E)$ and an integer $k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter:</td>
<td>$k$</td>
</tr>
<tr>
<td>Question:</td>
<td>Does $G$ have a vertex cover of size $k$?</td>
</tr>
</tbody>
</table>

**Algorithms for Vertex Cover**

- brute-force: $O^*(2^n)$
- brute-force: $O^*(n^k)$
- vc1: $O^*(2^k)$ (cf. Lecture 1)
- vc2: $O^*(1.4656^k)$ (cf. Lecture 1)
- fastest known: $O(1.2738^k + k \cdot n)$ [Chen, Kanj, Xia, 2010]
Running times in practice

\( n = 1000 \) vertices, \( k = 20 \) parameter

<table>
<thead>
<tr>
<th>Theoretical</th>
<th>Running Time</th>
<th>Nb of Instructions</th>
<th>Real</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 2^n )</td>
<td></td>
<td>( 1.07 \cdot 10^{301} )</td>
<td>4.941 \cdot 10^{282} ) years</td>
</tr>
<tr>
<td>( n^k )</td>
<td></td>
<td>( 10^{60} )</td>
<td>4.611 \cdot 10^{41} ) years</td>
</tr>
<tr>
<td>( 2^k \cdot n )</td>
<td></td>
<td>( 1.05 \cdot 10^9 )</td>
<td>15.26 milliseconds</td>
</tr>
<tr>
<td>( 1.465^k \cdot n )</td>
<td></td>
<td>( 2.10 \cdot 10^6 )</td>
<td>0.31 milliseconds</td>
</tr>
<tr>
<td>( 1.2738^k + k \cdot n )</td>
<td></td>
<td>( 2.02 \cdot 10^4 )</td>
<td>0.0003 milliseconds</td>
</tr>
</tbody>
</table>

Notes:
- We assume that \( 2^{36} \) instructions are carried out per second.
- The Big Bang happened roughly \( 13.5 \cdot 10^9 \) years ago.

Goal of Parameterized Complexity

Confine the combinatorial explosion to a parameter \( k \).

(1) Which problem–parameter combinations are fixed-parameter tractable (FPT)? In other words, for which problem–parameter combinations are there algorithms with running times of the form

\[ f(k) \cdot n^{O(1)}, \]

where the \( f \) is a computable function independent of the input size \( n \)?

(2) How small can we make the \( f(k) \)?

Examples of Parameters

<table>
<thead>
<tr>
<th>A Parameterized Problem</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input: an instance of the problem</td>
</tr>
<tr>
<td>Parameter: a parameter</td>
</tr>
<tr>
<td>Question: a YES–NO question about the instance and the parameter</td>
</tr>
</tbody>
</table>

- A parameter can be
  - solution size
  - input size (trivial parameterization)
  - related to the structure of the input (maximum degree, treewidth, branchwidth, genus, ...)
  - combinations of parameters
  - etc.

1.2 Coloring

A \( k \)-coloring of a graph \( G = (V, E) \) is a function \( f : V \rightarrow \{1, 2, ..., k\} \) assigning colors to \( V \) such that no two adjacent vertices receive the same color.

<table>
<thead>
<tr>
<th>COLORING</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input: Graph ( G ), integer ( k )</td>
</tr>
<tr>
<td>Parameter: ( k )</td>
</tr>
<tr>
<td>Question: Does ( G ) have a ( k )-coloring?</td>
</tr>
</tbody>
</table>
Brute-force: $O^*(k^n)$, where $n = |V(G)|$. Inclusion-Exclusion: $O^*(2^n)$. FPT?

**Coloring is probably not FPT**

- Known: Coloring is NP-complete when $k = 3$
- Suppose there was a $O^*(f(k))$-time algorithm for Coloring
  - Then, 3-Coloring can be solved in $O^*(f(3)) \subseteq O^*(1)$ time
  - Therefore, P = NP
- Therefore, Coloring is not FPT unless P = NP

### 1.3 Clique

A *clique* in a graph $G = (V, E)$ is a subset of its vertices $S \subseteq V$ such that every two vertices from $S$ are adjacent in $G$.

**Clique**

<table>
<thead>
<tr>
<th>Input:</th>
<th>Graph $G = (V, E)$, integer $k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter:</td>
<td>$k$</td>
</tr>
<tr>
<td>Question:</td>
<td>Does $G$ have a clique of size $k$?</td>
</tr>
</tbody>
</table>

Is Clique NP-complete when $k$ is a fixed constant? Is it FPT?

**Algorithm for Clique**

- For each subset $S \subseteq V$ of size $k$, check whether all vertices of $S$ are adjacent
- Running time: $O^* \left( \binom{n}{k} \right) \subseteq O^*(n^k)$
- When $k \in O(1)$, this is polynomial
- But: we do not currently know an FPT algorithm for Clique
- Since Clique is W[1]-hard, we believe it is not FPT. (See lecture on W-hardness.)

### 1.4 Δ-Clique

**A different parameter for Clique**

<table>
<thead>
<tr>
<th>Δ-Clique</th>
<th>Input:</th>
<th>Graph $G = (V, E)$, integer $k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter:</td>
<td>$\Delta(G)$, i.e., the maximum degree of $G$</td>
<td></td>
</tr>
<tr>
<td>Question:</td>
<td>Does $G$ have a clique of size $k$?</td>
<td></td>
</tr>
</tbody>
</table>
Is $\Delta$-Clique FPT?

Algorithm for $\Delta$-Clique

Input: A graph $G$ and an integer $k$.
Output: Yes if $G$ has a clique of size $k$, and No otherwise.

if $k = 0$ then
  return Yes
else if $k > \Delta(G) + 1$ then
  return No
else
  /* A clique of size $k$ contains at least one vertex $v$.
     For each $v \in V$, we check whether $G$ has a $k$-clique $S$ containing $v$ (note that
     $S \subseteq N_G[v]$ in this case). */
  foreach $v \in V$ do
    foreach $S \subseteq N_G[v]$ with $|S| = k$ do
      if $S$ is a clique in $G$ then
        return Yes
  return No

Running time: $O^*((\Delta + 1)^k) \subseteq O^*((\Delta + 1)^\Delta)$. (FPT for parameter $\Delta$)

2 Basic Definitions

Main Parameterized Complexity Classes

$n$: instance size
$k$: parameter

P: class of problems that can be solved in $n^{O(1)}$ time
FPT: class of parameterized problems that can be solved in $f(k) \cdot n^{O(1)}$ time
XP: class of parameterized problems that can be solved in $f(k) \cdot n^{g(k)}$ time (“polynomial when $k$ is a constant”)

$P \subseteq FPT \subseteq W[1] \subseteq W[2] \cdots \subseteq W[P] \subseteq XP$

Known: If $FPT = W[1]$, then the Exponential Time Hypothesis fails, i.e. 3-Sat can be solved in $2^{o(n)}$ time, where $n$ is the number of variables.

Note: We assume that $f$ is computable and non-decreasing.

3 Further Reading