Vertex Covers: Indirect Certificates and New FPT Algorithms

CAI Leizhen

CSE, CUHK
Outline

- Introduction
- Indirect certificates
- FPT algorithms
- Conclusion
Introduction

Vertex Cover (NP-complete)
Input: Graph $G = (V,E)$, parameter $k$.
Question: Does $G$ contain $k$ vertices that cover all edges?
Task: Compute $2^n$

Direct: $O(n^2)$ time.
Repeated squaring: $O(n^{1.59})$ or $O(n \log^2 n \log \log n)$ time.

Input size: $O(\log n)$.
Question: Can we do it in polynomial time?
Answer: No, because output size $\Theta(n)$. 
Parameterized Complexity

Input $I \rightarrow$ Algorithm $A \rightarrow$ Output $O$

Time complexity: classical $T(|I|) \rightarrow$ 2D-way $T(|I|, |O|)$

Parameterized complexity: $T(|I|, k)$

$k$: parameter of interest, typically $|O|$, solution size, or structural parameter (e.g., number of edge deletions to obtain a planar graph).
FPT Algorithms

FPT algorithm: $f(k)n^{O(1)}$ time.

$FPT = \text{fixed-parameter tractable}$

$k^k n$

$4^k k^2 n^2$

$2^k n$

$2^{\sqrt{k}} n$

$n^2 + 2^k$

$1.2738^k + kn$

To solve NP-hard problems effectively for relatively small $k$. 
## FPT Algorithms

<table>
<thead>
<tr>
<th></th>
<th>$k$</th>
<th>10</th>
<th>20</th>
<th>50</th>
<th>100</th>
<th>$n = 1000$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DS</strong></td>
<td>$n^k$</td>
<td>$10^{30}$</td>
<td>$10^{60}$</td>
<td>$10^{150}$</td>
<td>$10^{300}$</td>
<td></td>
</tr>
<tr>
<td><strong>IS</strong></td>
<td>$n^{0.8k}$</td>
<td>$10^{24}$</td>
<td>$10^{48}$</td>
<td>$10^{120}$</td>
<td>$10^{240}$</td>
<td></td>
</tr>
<tr>
<td><strong>VC</strong></td>
<td>$2^k n$</td>
<td>$10^{6}$</td>
<td>$10^{9}$</td>
<td>$10^{18}$</td>
<td>$10^{33}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$1.2738^k + kn$</td>
<td>$10^{4}$</td>
<td>$10^{4}$</td>
<td>$10^{5}$</td>
<td>$10^{10}$</td>
<td></td>
</tr>
</tbody>
</table>

Vertex Cover, Clique, Independent Set:

No problem to obtain optimal solutions for graphs with 200 vertices!
Vertex Cover

Input: Graph $G = (V,E)$, parameter $k$.
Question: Does $G$ contain $k$ vertices that cover all edges?

Task: FPT algorithms for Vertex Cover.
FPT Algorithms for Vertex Cover

Graph minor

Fellow and Langston (1986) \( \rightarrow O(f(k)n^3) \)
f(k) astronomical

Johnson (1987) \( \rightarrow O(f(k)n^2) \)
f(k) \( \approx 2^{2500k} \)

Matching

Papadimitriou and Yannakakis (1993) \( \rightarrow O(3^k kn) \)
FPT algorithms for Vertex Cover

Bounded search tree

- For any edge $uv$, either $u$ or $v$ must be in a solution $\rightarrow O(2^k kn)$
- Path $P_3$ $\rightarrow O(1.618^k kn)$
- Vertex of degree at least 3 $\rightarrow O(1.5^k kn)$
- Chan, Kanj, and Xia (2010) $\rightarrow O(1.2738^k + kn)$
Ways to Finish 100M

倒走 拿大顶 飞滚 旋翻 单腿跳 滑雪型转步
西施步 玉环醉酒 扭臀步 太空漫游
凌波虚步 精神病人思路广 猫行 梦游 旋风腿
僵尸跳 济公步 比翼双飞 秧歌摆 小鲜肉步
开车 租人 趟泥步 倒撵猴 乘火箭 喷 打的
快闪 最少能 弹弓 风火轮 交叉迴旋

March 2, 2019
IC-LYCS 2019, Okinawa, Japan
Introduction: Motivations

大道至简 Greatest truth is simple

- Better understanding
- Training students
- Intellectually challenging

化腐朽为神奇 Do bad things in clever ways
New FPT Algorithm for Vertex Cover

Randomly mark each vertex, output N(M).

M: marked vertices.
N(M): neighbors of marked vertices.
Certificate for Vertex Cover

Vertex Cover belongs to class NP.

Natural certificate: solution, i.e., a $k$-vertex cover $X$.

Alternative certificate: subset of $X$ with at most $k - \log n$ vertices.
Theorem 1. For any minimal vertex cover $X$ of a graph $G = (V,E)$, $V - X$ contains at most $|X|$ vertices $C$ such that $N(C) = X$. 

Indirect Certificate
Indirect Certificate

[Diagram of a network with interconnected nodes]

March 2, 2019

IC-LYCS 2019, Okinawa, Japan
Indirect Certificate

Given indirect certificate C, we can obtain vertex cover X in linear time.

Therefore C can be used as a certificate to verify that G indeed has a $k$ vertex cover.
FPT Algorithm using Indirect Certificate

Partition vertices of $G$ into blue vertices $B$ and red vertices $R$ such that
- $B$ contains vertex cover $X$, and
- $R$ contains indirect certificate $C$.

Once we have such a $(B,R)$-partition, Theorem 1 guarantees that $N(R)$ is a required vertex cover.

How to produce such a $(B,R)$-partition?
Randomized FPT Algorithm

Algorithm VC-IC

Step 1. Randomly and independently color each vertex either red or blue with probability $\frac{1}{2}$ to form red vertices $R$.

Step 2. Return $N(R)$ as a solution.

Theorem 2. Algorithm VC-IC finds, with probability at least $4^{-k}$, a $k$-vertex cover of $G$, if it exists, in $O(m + n)$ time.

Note: The algorithm can be derandomized by $(n, 2k)$-universal sets.
Semi-random Partition

Repeat the following until all vertices of G are coloured:

- Randomly choose an uncoloured vertex \( v \), colour it red or blue with probability \( p \) for red and probability \( 1-p \) for blue, and

- colour all neighbours of \( v \) blue if \( v \) is coloured red.
Random Selection

Optimal value for $p$ is 1.

Randomly choose a vertex $v$ and declare it to be not in solution, and hence put all vertices of $N(v)$ into solution.
Random Selection

Algorithm VC-SRP

Step 1. Repeat the following until all vertices are coloured:

Randomly and uniformly choose an uncoloured vertex $v$, colour $v$ red and all neighbours of $v$ blue to form a (B,R)-partition of $V$.

Step 2. Output $N(R)$ as $X$.

Theorem 3. Algorithm VC-SRP finds, with probability at least $2^{-k}$, a $k$-vertex cover of $G$, if it exists, in $O(m + n)$ time.
Theorem 4. Every yes-instance \((G,k)\) of Vertex Cover admits an indirect certificate \(C\) with at most \(k/3\) vertices.

Success probability better

\[
\text{VC-IC}: 2.1166^{-k} \\
\text{VC-SRP}: 1.6633^{-k}
\]
Smaller Indirect Certificate
Conclusion

- Indirect certificates are interesting in their own right.
- Potential to use indirect certificates to obtain FPT algorithms.