Vertex Cover is Fixed-Parameter Tractable

CS 511

Iowa State University

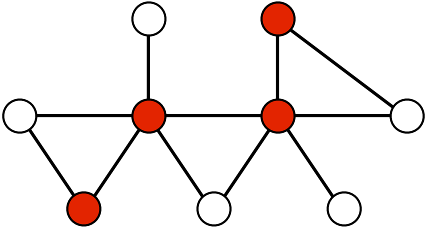
November 28, 2010

The Vertex Cover Problem

Vertex Cover

Input: An undirected graph $G = (V, E)$ and an integer k.

Problem: Is there a subset of vertices $S \subseteq V$ such that $|S| \leq k$, and for each edge $(u, v) \in E$ either $u \in S$, or $v \in S$, or both?



Theorem

Vertex Cover is NP-complete.

What if k is small?

- **Brute force algorithm:** $O(kn^{k+1})$.
  - Try all $\binom{n}{k} = O(n^k)$ subsets of size k.
  - Takes $O(kn)$ time to check whether a subset is a vertex cover.
- **Goal:** Limit exponential dependency on k to get a practical algorithm for small k.
- **Results**

Algorithm	Time(n, k)	Time$(10^4, 10)$	Feasible?
Brute force	kn^{k+1}	10^{45}	No
Better	$2^k kn$	10^8	Yes
Kernelized	$2^k k^2 + kn$	$2 \cdot 10^5$	Yes!

If k is small, G must be sparse

Lemma (Bound on the number of edges)

If G has a vertex cover of size k, it has $\leq k(n-1)$ edges.

Proof.

Each vertex can cover at most $n-1$ edges, so k vertices can cover at most $k(n-1)$ edges. □

Every edge must be covered by one of its endpoints

Lemma (Edge coverage)

Let (u, v) be an edge of G. G has a vertex cover of size $\leq k$ iff at least one of $G - u$ and $G - v$ has a vertex cover of size $\leq k - 1$.

Proof.

$\Rightarrow$

- Suppose G has a vertex cover S of size $\leq k$.
- S contains either u or v (or both). Assume it contains u.
- $S \setminus \{u\}$ is a vertex cover of $G - u$.

$\Leftarrow$

- Suppose S is a vertex cover of $G - u$ of size $\leq k - 1$.
- Then $S \cup \{u\}$ is a vertex cover of G.

□

A Fixed-Parameter Algorithm for Vertex Cover

boolean Vertex-Cover(G, k) :

if (G contains no edges) **return** true
if (G contains $\geq kn$ edges) **return** false

let (u, v) be any edge of G

$a =$ Vertex-Cover$(G - u, k - 1)$
$b =$ Vertex-Cover$(G - v, k - 1)$

return $a \vee b$

Analysis

Theorem

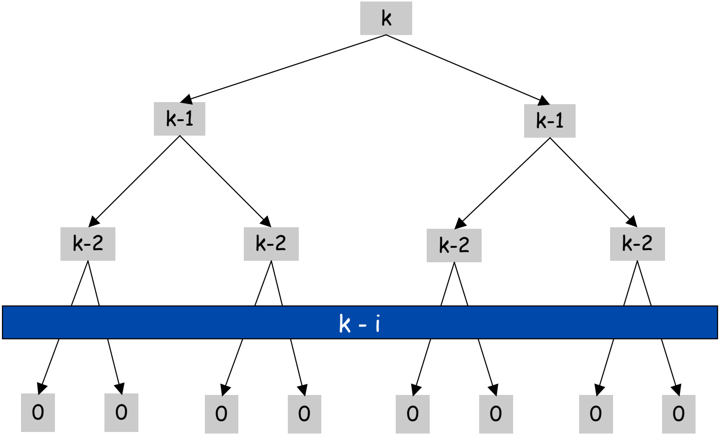
Algorithm Vertex-Cover determines if G has a vertex cover of size $\leq k$ in $O(2^k kn)$ time.

Proof.

- Correctness follows from the bound on the number of edges and the edge coverage lemma.
- There are $\leq 2^{k+1}$ nodes in the recursion tree; each invocation takes $O(kn)$ time.

□

Recursion Tree



Data Reduction

Observation

Suppose G has a vertex cover S of size $\leq k$ and that G contains a vertex v of degree $> k$. Then, $v \in S$.

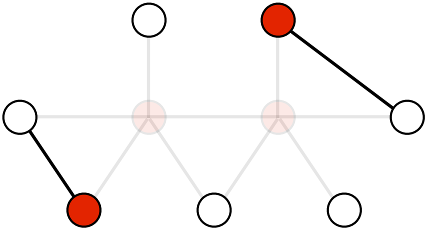
Proof.

If $v \notin S$, then each of v's at least $k + 1$ neighbors would have to be in S, a contradiction. □

Data Reduction

Corollary (Modified edge coverage lemma)

Let U be the set of all nodes of degree $> k + 1$ in G. Then G has a vertex cover of size $\leq k$ iff $G - U$ has a vertex cover of size $\leq k - |U|$.



$G - U$ is called the kernel of G (with respect to k-Vertex Cover).

Data Reduction

Lemma (Bound on the number of edges in the kernel)

If G has a vertex cover of size k, then G's kernel has $\leq k^2$ edges.

Proof.

Each vertex in the kernel can cover at most k edges, so k of its vertices can cover at most k^2 edges. □

An Improved Fixed-Parameter Algorithm

boolean Faster-Vertex-Cover(G, k) :

let U be the set of all nodes of degree $\geq k + 1$ in G
$G = G - U$
$k = k - |U|$
if (G has more than k^2 edges) **return** false
return Vertex-Cover($G - U, k - |U|$)

Analysis

Theorem

Algorithm Faster-Vertex-Cover determines if G has a vertex cover of size $\leq k$ in $O(2^k k^2 + nk)$ time.

Proof.

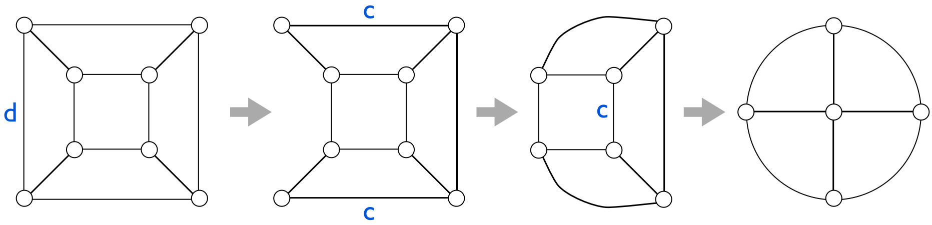
- Correctness follows from the correctness of algorithm Vertex-Cover, the bound on the number of edges in the kernel, and the modified edge coverage lemma.
- The running time follows from that of algorithm Vertex-Cover and the fact that the input graph has at most k^2 edges.

□

Graph Minors

Definition

Graph H is a minor of graph G if H is isomorphic to a graph that can be obtained by zero or more edge contractions on a subgraph of G.



$G = Q_3$ $H = W_4$

Minor-Closed Families of Graphs

Definition

A family $\mathcal{F}$ of graphs is minor-closed if every minor of a graph in $\mathcal{F}$ also belongs to $\mathcal{F}$.

Some Minor-Closed Families

- Forests.
- Graphs with vertex cover of size at most k.
- Planar graphs.
- Graphs with feedback vertex set of size at most k.
- Graphs of treewidth at most k.

Forbidden Minors

Theorem (Forbidden Minor Theorem — Robertson & Seymour)

Let $\mathcal{F}$ is a minor-closed family of graph and let $\mathcal{S}$ be the set of graphs that are not in $\mathcal{F}$. Then, there exists a finite set $\mathcal{H}$ of minimal elements in $\mathcal{S}$.

- The members of $\mathcal{H}$ are the excluded minors (or forbidden minors, or minor-minimal obstructions) for $\mathcal{F}$.
- $\mathcal{H}$ gives a forbidden graph characterization of $\mathcal{F}$.
  - $\mathcal{F}$ is precisely the set of graphs that do not have any graph in $\mathcal{H}$ as a minor.

Finding Forbidden Minors

Theorem (Finding Minors — Robertson & Seymour)

For each fixed graph H, there is a $O(n^3)$ algorithm to test whether an n-vertex graph G has H as a minor. The constant factor in the running time depends only on H

Thus, for every minor-closed family $\mathcal{F}$, there is polynomial time algorithm to test whether a graph G belongs to $\mathcal{F}$:

- For each forbidden minor H of $\mathcal{F}$, check whether G contains H.

Remarks

- The proof of the Forbidden Minor Theorem is non-constructive: It does not provide the obstruction set.
- The constant factor in the Minor-Finding Theorem depends superpolynomially on the size of the excluded minors.
- Robertson & Seymour's theorems are nevertheless useful.
  - Show the existence of a polynomial-time solution.
  - Suggest that a practical algorithm might exist.
  - May not even need the obstruction set.
  - E.g., Vertex Cover.