Parameterized Complexity – Kernelization

Serge Gaspers[1]

[1]CMM, University of Chile, Santiago, Chile

CC61X: Diseño y Analisis de Algoritmos Adaptivos

Outline

The Basics

A Parameterized Problem

- Input: an instance of the problem
- Parameter: a parameter k
- Question: a Yes–No question about the instance and the parameter

P: class of problems that can be solved in time $n^{O(1)}$
FPT: class of problems that can be solved in time $f(k) \cdot n^{O(1)}$
$W[\cdot]$: parameterized intractability classes

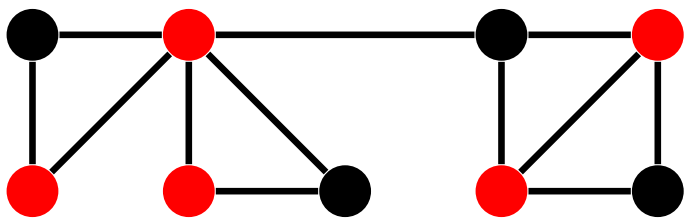
$$P \subseteq FPT \subseteq W[1] \subseteq W[2] \cdots \subseteq W[P]$$

↑
Vertex Cover

Vertex Cover

Vertex Cover (VC)

- Input: A graph $G = (V, E)$ on n vertices, an integer k.
- Parameter: k
- Question: Is there a set of vertices $C \subseteq V$ of size at most k such that every edge has at least one endpoint in C?

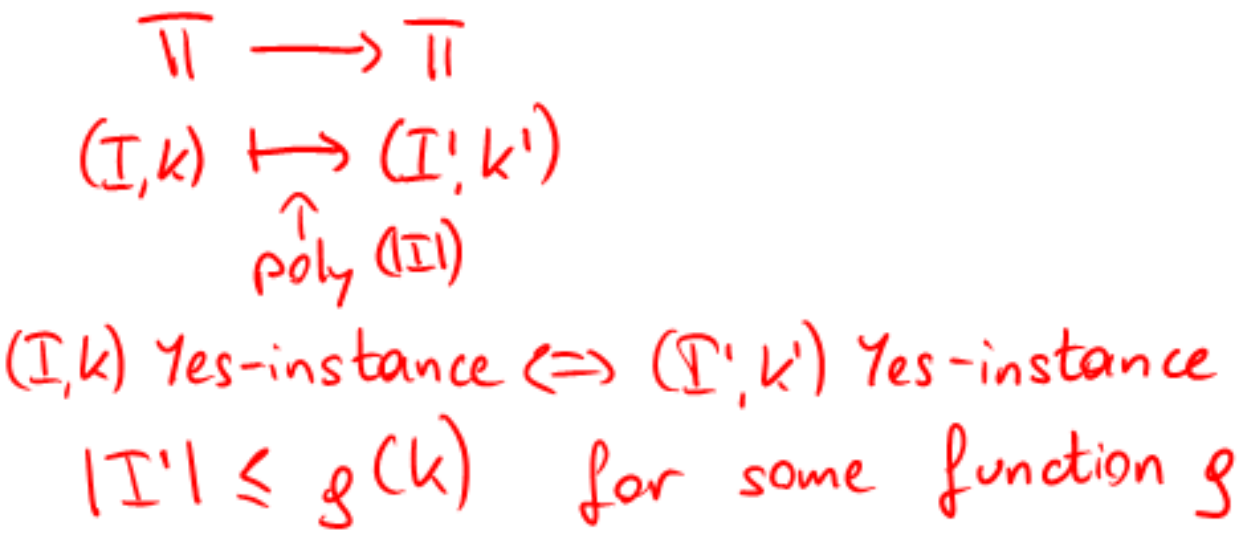


Outline

Kernelization Algorithm: definition

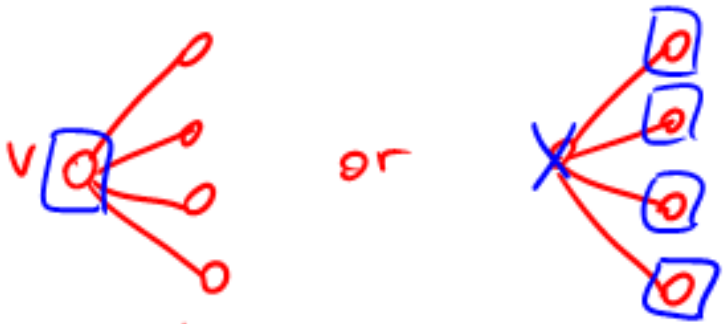
Definition 1 (Kernelization Algorithm)

A kernelization algorithm is a polynomial time algorithm, taking as input an instance (I, k) of a parameterized problem, and returning an equivalent instance (I', k') of the same problem such that $|I'| \leq g(k)$.
We say that the problem has a $g(k)$-kernel.



Outline

VC has a $(k \cdot (k+1))$-kernel



- $d(v) \geq k+1$
  → every k-v.c. contains v
  ⇝ $(G-v, k-1)$

$\leq k \cdot (k+1)$

- $d(v) = 0$
  → no min. v.c. contains v
  ⇝ $(G-v, k)$

If G contains more than $k \cdot (k+1)$ vertices
→ return NO
"trivial No-instance $(\circ\!\!-\!\!\circ, 0)$"

⇒ $k \cdot (k+1)$ - kernel

Outline

A problem is FPT iff it has a kernel

Theorem 2

Let Π be a decidable parameterized problem.
$\Pi \in$ FPT $\iff$ Π has a $g(k)$-kernel for some function g.

Proof (⇐) Reduce to kernel + brute-force algo.
$O(n^c)$ + $f(k)$

(⇒) ∃ $f(k) \cdot n^c$-time algo.
If $|I| \geq f(k)$, then run algo : $f(k) \cdot n^c \leq n^{c+1}$
and output trivial Yes/No-instance
If $|I| < f(k)$, then output (I, k)
↙
$f(k)$-kernel

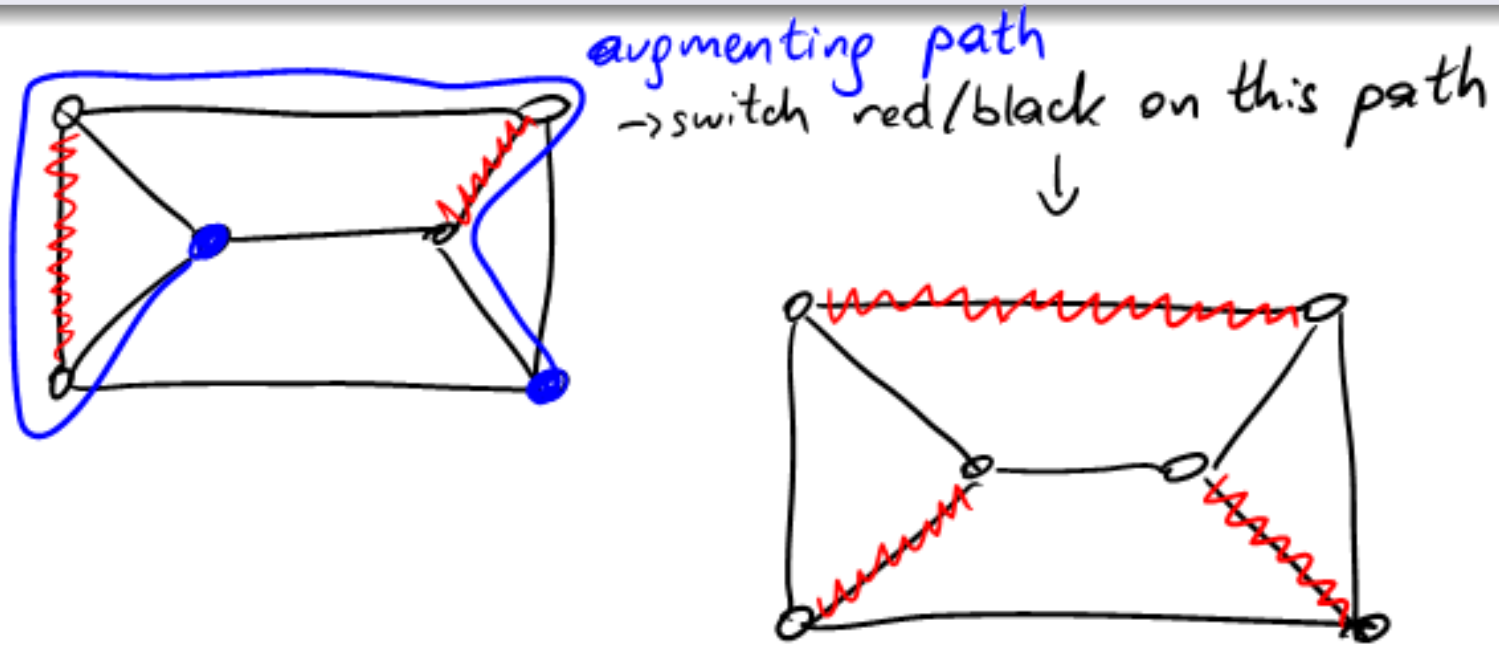
□

Outline

Matching: Definition and Example

Definition 3 (Matching)

A matching M in a graph is a set of edges such that no two edges of M share an endpoint.



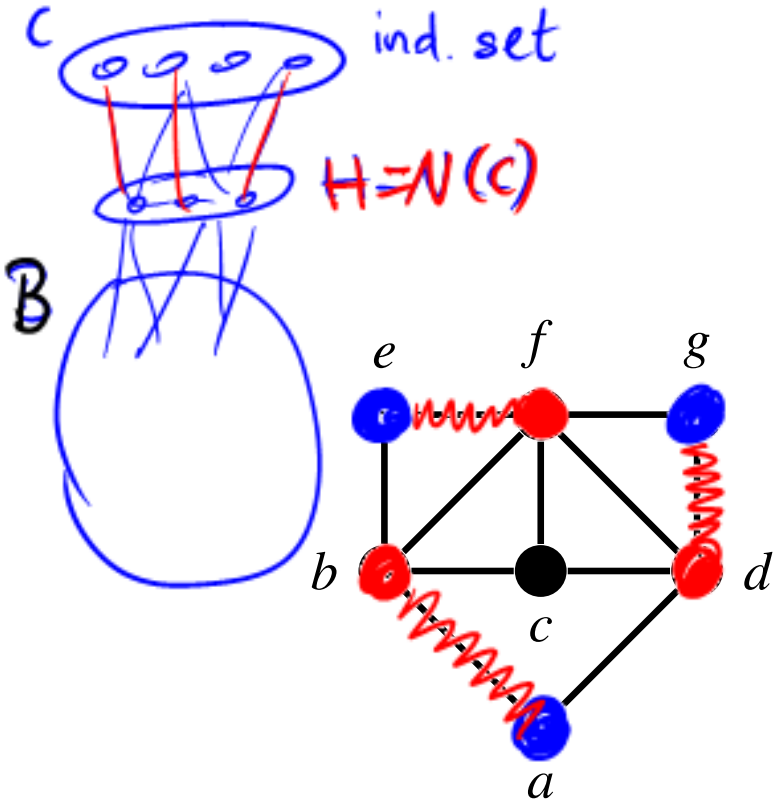
Crown Decomposition: Definition

Definition 4 (Crown Decomposition)

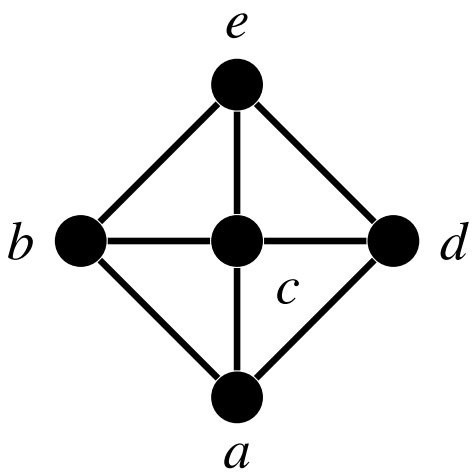
A crown decomposition (C, H, B) of a graph $G = (V, E)$ is a partition of V into sets C, H, and B such that

- the crown C is a non-empty independent set,
- the head H contains all the neighbors of the vertices in C,
- the body B contains the remaining vertices $V \setminus N[C]$, and
- there is a matching of size $|H|$ in $G[H \cup C]$.

Crown Decomposition: Examples



has crown dec.



has no crown dec.

Outline

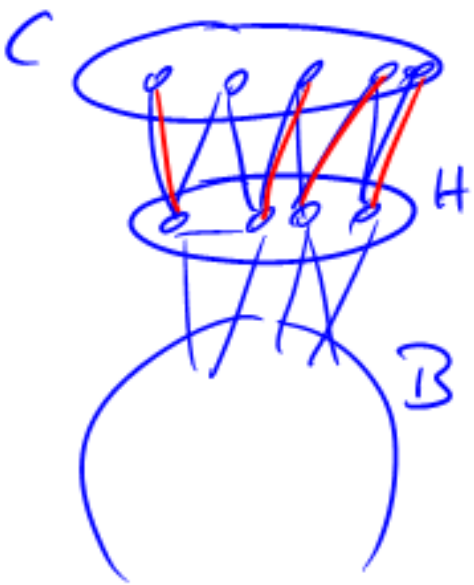
VC and Crown Decomposition

Lemma 5

Suppose that $G = (V, E)$ has a crown decomposition (C, H, B).

$$vc(G) \leq k \iff vc(G[B]) \leq k - |H|,$$

where $vc(G)$ denotes the size of the smallest vertex cover of G.



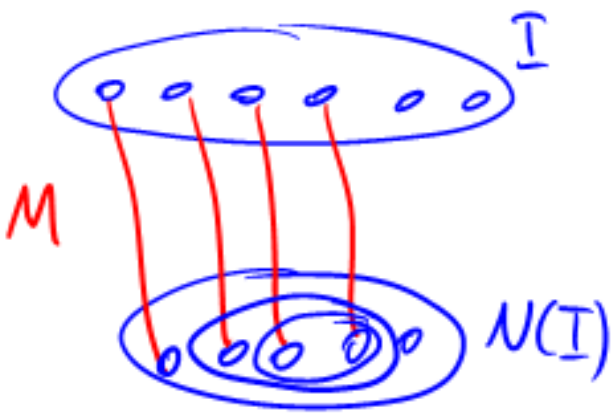
v.c. has ≥1 vertex from every edge of the matching
we need ≥ |H| vertices to cover edges in G[C∪H]
→ optimal to add H to the v.c.

↝ (G − H − C, k − |H|)

Finding a Crown Decomposition

Lemma 6

If $I \subseteq V$ is an independent set in G such that $|N(I)| < |I|$, then a crown decomposition (C, H, B) of G can be found in linear time.



$V(M) :=$ end points in M

$C^0 := N(I \setminus V(M))$

$C^i := N\left(N_M(C^{i-1})\right)$

$C := C^\infty$ ← nbs in (V, M)

$\left(N_M(C) \cup (I \setminus V(M)), C, V \setminus \right)$

[Chor, Fellows, Juedes -WG'04]

Outline

Vertex Cover has a linear kernel

$M :=$ maximum matching of G [Edmonds]

If $|M| > k$, answer No

$I := V \setminus V(M)$

If $|I| \leq 2k$, output (G, k)

$(C, H, B) :=$ crown dec. of G

$\rightsquigarrow$ recurse with $(G[B], k - |H|)$

$\Rightarrow$ $4k$-kernel for V.C.