CSC791/495-011

Dr. Blair D. Sullivan

August 25, 2017

Recap

github.com/bdsullivan/ParameterizedAlgorithms-Fall2017

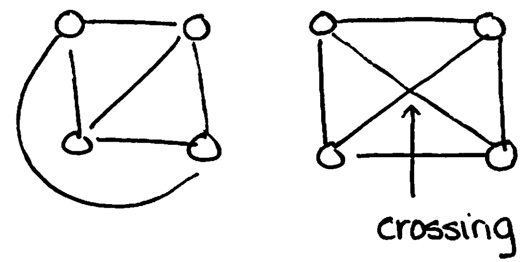
- This class is about:
  - having fun.
  - solving NP-complete problems <u>efficiently</u> & <u>exactly</u>
  - proofs & communication
  - research (in TCS)
- We designed 2 flavors of algorithms $(\overbrace{n^{f(k)}}^{XP}, \overbrace{f(k)n^c}^{FPT})$ to solve

  VERTEX COVER, COLORING, ☹ CLIQUE $\leftarrow n^k$

  ☹ NO FPT/XP
- There's a (relatively) large toolbox of methods – we saw examples of
  1. reduction / preprocessing / kernelization
  2. branching / bounded search tree
- Not all problems are susceptible to these approaches
- Promised an analogue of NP-hardness & complexity for parameterized setting.

Planar Independence

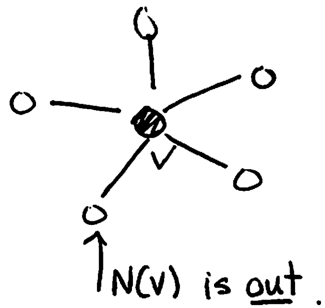
Problem: Given a planar graph G and $k \in \mathbb{Z}^+$, is there an independent set of size k in G?



Lemma If G is a planar graph, G has a vertex of $\deg \leq 5$.

Proof use Euler's formula $n - m + f = 2$

in planar $m \leq 3n - 6$ (use Δ faces)



If $v \in I$ (indep. set) $\Rightarrow N(v) \cap I = \emptyset$

Reduction: pick a vertex of $\deg \leq 5$. Add it to I. Delete $N[v]$. $k \to k-1$

Observe: if we reach $k=0$, we win (YES-instance). This always happens if $k < n/6 \; = \; n > 6k$

How big is the kernel? If "Reduction Rule" doesn't apply $\Rightarrow n \leq 6k$

strategy

this is an $O(k)$ Kernel.

Formal Definitions

Parameterized Problems, FPT, XP

(Σ fixed finite alphabet)

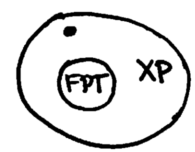
Defn A parameterized problem is $L \subseteq \Sigma^* \times \mathbb{N}$ w/ instance (x, k)

We call k the parameter and define the size of an instance to be $|x| + k$

Defn L is fixed parameter tractable if $\exists$ algorithm A, computable function $f: \mathbb{N} \to \mathbb{N}$, and constant c s.t. A (correctly) decides if $(x,k) \in L$ in time $f(k)\,|(x,k)|^c$

FPT is the class of all fixed parameter tractable problems.

Defn L is slice-wise polynomial if $\exists$ algorithm A, computable functions $f, g: \mathbb{N} \to \mathbb{N}$ s.t. A decides L in time $\leq f(k)\,|(x,k)|^{g(k)}$.



XP is the class of all slice-wise polynomial problems.

Note: $O^*(f(k))$ means running time $f(k)\,n^{O(1)}$

Kernelization

Defn a reduction rule is $\phi : \underbrace{\Sigma^* \times \mathbb{N}}_{\substack{\text{instance} \\ (I,k)}} \to \underbrace{\Sigma^* \times \mathbb{N}}_{\substack{\text{instance} \\ (I',k')}}$

where

1. (I,k) & (I',k) are equivalent instances $\quad (I,k) \in L \iff (I',k') \in L$
2. ϕ computable in poly$(|I|, k)$

a rule is "safe" if ① holds.

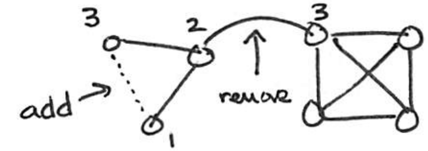
Defn a kernelization algorithm (kernel) is an algorithm A s.t.

1. polynomial time
2. produces (I',k') equivalent to (I,k)
3. $\exists$ computable function $g(\cdot)$ s.t. $|I'| + k' \leq g(k)$

We say the problem admits a polynomial (linear) kernel if g is polynomial (linear).

In-Class Exercise

this week's proof writing exercise



$k=1$ NO

$k=2$ YES

Defn G is a cluster graph if every connected component of G is a clique.

Problem: Given a graph G and $k \in \mathbb{Z}^+$, is there a set of at most k edge edits (add or delete a single edge in G) that make G into a cluster graph?

Thm ClusterEditing has an $O(k^2)$-vertex kernel.

Helpful Lemma G is a cluster graph $\Leftrightarrow$ G has no induced P_3.

Design reduction rules: suggestion: if o—o exists, it's a clique component $\Rightarrow$ delete it

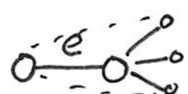
? is this the most general rule of this flavor that works?

works for any clique component.

Since cluster $\Leftrightarrow$ no induced P_3 $\Rightarrow$ editing only needs to elminate induced P_3's.

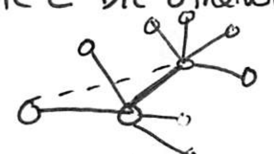
RR1 delete any vertex not in an induced P_3

Idea



want to delete e b/c otherwise we have to add/delete $\geq \deg(v)$ edges

concern:



! hard to show safe

Use vertex cover strategy: if some edge is in $> k$ induced P_3's $\Rightarrow$ it is in all edit sets of size $\leq k$.

⋮

Kernels are (the) Key

Thm A parameterized problem Q is FPT $\iff$ Q admits a kernel

Proof ($\Leftarrow$) given a kernel of size $g(k)$, I can decide Q by running any algorithm (say w/ running time $f(\cdot)$) on the kernel. This takes

↑ computable

$$f(g(k)) \cdot \underbrace{\mathrm{poly}(|I|, k)}_{\text{compute kernel (defn)}} \longleftarrow f'(k)\,|(I,k)|^{O(1)} \quad \underline{\text{FPT}}$$

($\Rightarrow$) $\exists\, A$ runs in $\underline{f(k)\,|(I,k)|^c}$ (defn of FPT).

My kernel: run A for $|(I,k)|^{c+1}$ steps. } polynomial-time

If A terminated w/ a decision, return a trivial YES or NO instance to match.

If not, return (I,k). But $f(k)\,|(I,k)|^c > |(I,k)|^{c+1}$ (otherwise A finishes)

$\Rightarrow |(I,k)| < f(k)$. ∎ [this is TERRIBLE!]

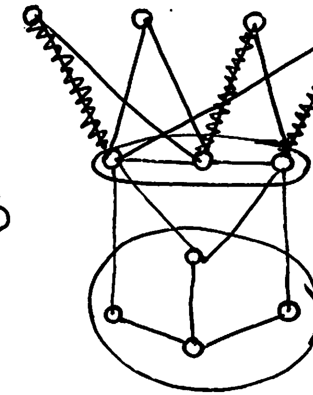
Lemmata

<u>König's Thm</u> In every undirected bipartite graph, the size of a maximum matching is equal to the size of a minimum vertex cover.

<u>Hall's Thm</u> Let G be an undirected bipartite graph with bipartition (V_1, V_2). Then G has a matching saturating $V_1 \iff |N(X)| \geq |X| \ \forall \ X \subseteq V_1$.

<u>Hopcroft-Karp Alg.</u> G an undirected bipartite graph w/ bipartition (V_1, V_2). Then one can find a maximum matching & minimum vertex cover in $O(m\sqrt{n})$.

Crown Decompositions

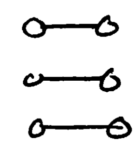


Defn a crown decomposition is partition of $V(G)$ into C, H, B

1. $C \neq \emptyset$ and an independent set
2. H separates C and B (there are no C-B edges)
3. there is a matching of size $|H|$ between C and H

Thm (crown lemma) If G is isolate-free and $|V(G)| \geq 3k+1$ then in polynomial time, we can either

1. find a matching of size $\geq k+1$ in G



or

2. G has a crown decomposition.

Note: Idea for crown lemma in kernels is that it guarantees strong structure whenever max Matching is small (VC is small) : useful if parameter bounds these.

Dual Vertex Coloring

aka Saving k Colors

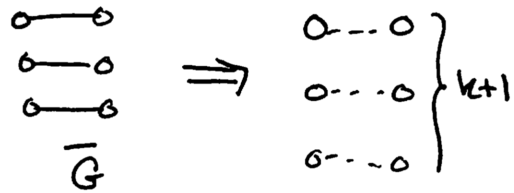
Problem Given a graph G and $k \in \mathbb{Z}^+$, can G be properly colored with $|V(G)|-k$ colors?

Thm Dual-Coloring has a linear kernel.

Key: look at $\overline{G}$

Key Lemma If $\overline{G}$ is isolate-free, either

① matching of $\geq k+1$



then I can color each pair w/ 1 color. ✓

so we can color w/ $n-(k+1)$

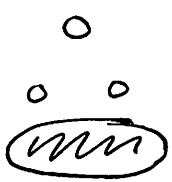
② $\overline{G}$ has a crown decomposition $\Rightarrow$ REDUCE!

③ $|V(\overline{G})| = |V(G)| < 3k+1 \Rightarrow$ KERNEL

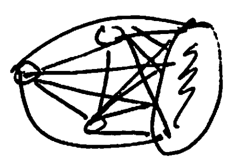
Need to (a) make $\overline{G}$ isolate-free

(b) show how to reduce in ②

If $\overline{G}$ has isolates:



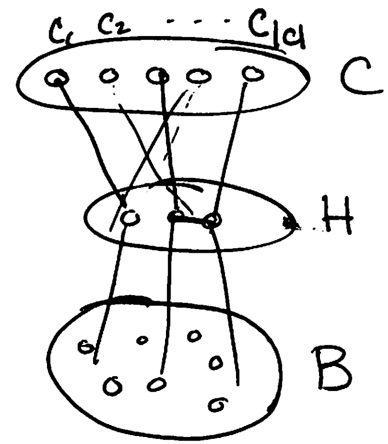
$\Rightarrow$ in G



these are apices so they must get a unique color.

Add a preprocessing rule to delete.

Crowns in Dual Coloring



given crown decomposition of $\overline{G}$

goal: color crown safely & delete

Q: what is special about C? Independent $\Rightarrow$ in G it's a clique.
so all must get different colors.

- can I also color H? yes b/c H is matched to C, we can color w/ same as other endpt of matching edge.

want to replicate apex argument — that these colors can't be re-used, so I can delete $C \cup H$ & make k smaller.

WIN H is a separator in $\overline{G}$ $\Rightarrow$ C is completely connected to B in G

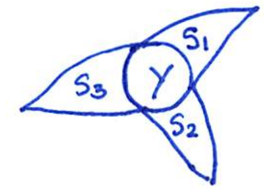
reduce $(G \setminus (H \cup C), \underline{\underline{k - |H|}})$

be careful

used $|C|$ colors
& made $n \rightarrow n - |C| - |H|$
$\Rightarrow k \rightarrow k - |H|$.

Sunflowers

Defn A sunflower with k petals and core Y is a collection of k sets $S_1, \ldots, S_k$ s.t. $S_i \cap S_j = Y \ \forall i \neq j$ and $S_i \setminus Y \neq \emptyset \ \forall i$.



(most pitiful sunflower ever!)

Thm (Erdös-Rado) Given a set family $\mathcal{A}$ over universe U with $|A| = d \ \forall A \in \mathcal{A}$, if $|\mathcal{A}| > d!(k-1)^d \Rightarrow \mathcal{A}$ contains a sunflower with k petals (& this is poly-time computable).

Proof Proceed by induction on d. This is a great exercise!

d-Set Packing

<u>Problem:</u> Given a family $\mathcal{A}$ of sets over universe $\mathcal{U}$ where $|A| \leq d \ \forall A \in \mathcal{A}$ and $k \in \mathbb{Z}^+$. Are there sets $S_1, \ldots, S_k \in \mathcal{A}$ that are pairwise disjoint?

<u>Show</u> d-Set Packing has an $f(d)k^d$-set kernel.

<u>Lemma</u> if $\mathcal{A}$ has a $(dk+1)$-sunflower $\Rightarrow$ we can create an equivalent instance $(\mathcal{A}', k)$ with $|\mathcal{A}'| < |\mathcal{A}|$.

<u>Proof</u> Consider the $(dk+1)$ petals. Pick an arbitrary petal P. We claim $(\mathcal{A} \setminus P, k)$ is an equivalent instance. ① If $\mathcal{A}$ has k pairwise disjoint sets $S_1, \ldots, S_k$ that don't include P $(S_i \neq P \ \forall i) \Rightarrow \mathcal{A} \setminus P$ does. ② If every family of prwise disjoint sets includes P, consider any one of them $(S_1, \ldots, S_k = P)$. Then $|\bigcup S_i| \leq dk$ (every set in $\mathcal{A}$ had size $\leq d$) but the sunflower has $dk+1$ petals $\Rightarrow$ some petal Q is disjoint from $S_1, \ldots, S_{k-1}$. Then $\{S_1, \ldots, S_{k-1}, Q\}$ is the desired family in $\mathcal{A} \setminus P$. ③ If $\mathcal{A}$ was a no-instance, deleting a set cannot make it a yes-instance.

<u>Algorithm / Kernel:</u> Argue similarly to d-Hitting Set in platypus book (consider $d' \in \{1, \ldots, d\}$ and argue if no reduction using lemma, $|\mathcal{A}|$ small).

Today's Problems

① Give a polynomial kernel for Minimum Maximal Matching (MMM):
undirected graph G, $k \in \mathbb{Z}^+$, is there a maximal matching of G on $\leq k$ edges?

② Use crown decompositions to give a linear kernel for Vertex Cover.

③ Use the sunflower lemma to give an $O(k^3)$ kernel for Cluster Vertex Deletion: undirected graph G, $k \in \mathbb{Z}^+$, is there a set $S \subseteq V(G)$ with $|S| \leq k$ s.t. $G \setminus S$ is a cluster graph?