Lecture 1 (22.02.2013) Author: Arkadiusz Socała

Introduction. Sunflower lemma.

1 Kernelization

1.1 Introduction

We are going to study kernelization as a way of preprocessing. We want to shrink our instance of the problem into some smaller instance (not necessarily a sub-instance of the original instance in any sense). We can consider this as a noise erasure. Our preprocessing should left only interesting (hard) part of the problem – a kernel.

Let's consider we have a poly-time algorithm A for some NP-hard problem that it always shrink an instance to 99% of its size. We see that such an algorithm should not exist. In other case we could iterate it and obtain a poly-time algorithm for the NP-hard problem. Similarly there should not exist a poly time algorithm always reducing size of the instance by one bit.

We study problems with a parameter. It means that for each instance of the problem we have some integer number. This number could be some measure of the instance. For example a treewidth could be a parameter for the problem which instances are graphs. Also often, when we want to find the largest object of some kind, the size of the solution is an interesting parameter (for example searching the longest path in a graph and the length as a parameter). We are interested in solving instances with small parameter.

Definition 1. Let's consider a parametrized language Q where $(x, k) \in \Sigma^* \times \mathbb{N}$. We say A is a *kernelization algorithm* for Q with *kernel* of size g when

(i) A transforms (x, k) (with k in unary format) into (x', k') in poly-time,

(ii) $(x, k) \in Q \iff (x', k') \in Q$,

(iii) $|x'|, k' \le g(k)$ and g is a computable function.

There exists also alternative, stronger definition where we demand $|x'| \le g(k)$ but $k' \le k$. In practice algorithms satisfying the first definition *often* satisfies also the latter.

Currently known lower bound techniques can show that there is no kernel for both definitions.

Definition 2. Language $L \subseteq \Sigma^*$ is *fixed parameter tractable* (FPT) for parameter $k : \Sigma^* \to \mathbb{N}$ if it can be solved in $\mathcal{O}(f(k(x)) \cdot poly(|x|))$ time, for some computable function f.

Theorem 3. *Let Q be a decideable parametrized language. Then $Q \in FPT$ if and only if there exists a kernelization algorithm for Q.*

Proof. Let's assume that A is a kernelization algorithm for Q that transforms (x, k) into (x', k') where $|x'|, k' \le g(k)$. Then we can use A in $poly(|x|, k)$ and then use any brutal solution on (x', k') in time bounded by a function of $g(k)$.

Now let's assume that we have a FPT algorithm for Q working in $f(k) \cdot |x|^c$. We can take $g = f$. Indeed, if $|x| \le f(k)$ then we can return just (x, k). In other case we have that $f(k) \cdot |x|^c \le |x|^{c+1}$ so we can solve the problem in poly-time $\mathcal{O}(n^{c+1})$. □

So our krenelization algorithm definition is another FPT algorithm definition. But this time we concentrate our studies on the preprocessing aspect of these algorithms.

We are interested in *small* functions g :

Definition 4. We say that a kernelization algorithm gives us a *polynomial kernel* if $g = poly(k)$. Similarly we say that kernelization algorithm gives us a *linear kernel* if $g = \mathcal{O}(k)$.

Obviously the class of problems with linear kernels is inside the class of problems with polynomial kernels. Both are inside the class of problems with an FPT algorithm. We have strong motivation to believe that both inclusions are not equalities.

1.2 Examples

1.2.1 Vertex Cover

We study the following problem.

VERTEX COVER **Input:** A graph G and an integer k **Question:** Does there exists a set $X \subseteq V(G)$ of size at most k, such that every edge of G has at least one endpoint in G?

We can easily give a bunch of simple ideas how to transform an instance of this problem. For example we can throw away isolated vertices. We have to take vertices with degree $\geq k+1$ to the solution. Also if there exists a leaf u with a neighbour v in our graph then there exits a solution containing v and not containing u.

We can take these ideas into more formal reduction rules:

(1) throw away isolated vertices,

(2) if we have v that $deg(v) > k$ then $G := G - v$ and $k := k - 1$,

(3) if rules (1) and (2) do not work and $|E(G)| > k^2$ then answer NO because after (2) each vertex can cover at most k edges.

So after these reductions we know that $|E| \leq k^2$ and $|V| \leq 2k^2$. These reductions are attributed to Buss [1].

There are better algorithms, yielding $\mathcal{O}(k)$ vertices, but probably we cannot get $k^{2-\varepsilon}$ edges [5].

1.2.2 Planar Independent Set

PLANAR INDEPENDENT SET **Input:** A planar graph G and an integer k **Question:** Does there exists in G an independent set of size k?

We know that in planar graph there exists a vertex of degree ≤ 5. So if $k \leq \frac{|V|}{6}$ then there exists a solution (we can find that solution by simple recursive algorithm). Let's formalize our reduction rule:

(1) if $|V| > 6k$ answer YES.

After applying this rule we know that $|V| \leq 6k$.

Note that we have used a 6-colour theorem. We could use a 4-colour theorem and get a better constant.

1.2.3 Edge Clique Cover

EDGE CLIQUE COVER **Input:** A graph G and an integer k **Question:** Find cliques $C_1, \dots, C_k \leq G$ such that $\bigcup_{i=1}^{k} E(C_i) = E(G)$.

We can also reformulate this question into: if there exists a family $\{A_v \subseteq \{1, \dots, k\} | v \in V(G)\}$ such that $uv \in E(G) \iff A_v \cap A_u \neq \emptyset$.

Let's take the reduction rules:

(1) remove isolated vertices,

(2) if we have an isolated edge uv then $G := G \setminus \{u, v\}$, $k := k - 1$

(3) if we have an edge uv and $N(u) \setminus \{v\} = N(v) \setminus \{u\}$ then $G := G \setminus v$.

If $C_1, \dots, C_k$ is a solution then a family defined that $i \in A_v \iff v \in C_i$ is a solution of the reformulated version of the problem.

After our reductions there is no v such that $A_v = \emptyset$ because it would be an isolated vertex. Also there is no u, v such that $A_u = A_v$ because then $N(u) \setminus \{v\} = N(v) \setminus \{u\}$. So sets of the family are pairwise different and then we can introduce a new rule:

(4) if rules (1)-(3) do not work and $|V| > 2^k$ then answer NO.

And we know that $|V| \leq 2^k$. These reductions are from [6].

There is known that obtaining $2^{o(k)}$ vertices is impossible under the assumption of Exponential Time Hypothesis [4]. Moreover $\mathcal{O}(k^c)$ vertices would cause a collapse of polynomial hierarchy [3].

2 Sunflower Lemma

Definition 5. Let $\mathcal{F} \subseteq 2^U$ for some set U. We define *s-sunflower* as $A_1, \dots, A_s \in \mathcal{F}$ such that $\forall_{1 \leq i < j \leq s} A_i \cap A_j = \bigcap_{r=1}^{s} A_r$.

Theorem 6 (Erdős-Rado 1960)**.** *Let $\mathcal{F} \subseteq 2^U$ for some set U such that $\forall_{A \in \mathcal{F}} |A| \leq d$ and $|\mathcal{F}| > d! \cdot k^d$. Then there exists a $(k+1)$-sunflower. Moreover it can be found in poly$(|\mathcal{F}|, d, k)$-time.*

Proof. Let's start with constructing a sunflower with disjoint elements of $\mathcal{F}$. We construct it in a greedy way and obtain some maximal sunflower $A_1, \dots, A_s$ with disjoint elements.

If $s > k$ then we are done.

If $s \leq k$ then for $C := A_1 \cup \ldots \cup A_s$ we know that $|C| \leq d \cdot k$ and $\forall_{A \in \mathcal{F}} A \cap C \neq \emptyset$. But $|\mathcal{F}| > d! \cdot k^d$ so there exits $v \in C$ that for $\mathcal{F}_v := \{A \in \mathcal{F} | v \in A\}$ we have $|\mathcal{F}_v| > (d-1)! \cdot k^{d-1}$.

Then we can assign $\mathcal{F}' := \{A - v | a \in \mathcal{F}_v\}$ and because $|\mathcal{F}'| > (d-1)! \cdot k^{d-1}$ then we can use an induction to get $(k+1)$-sunflower S' (the case for $d = 1$ is obvious). To get a desired $(k+1)$-sunflower we add v to every element of the S'. $\square$

2.1 Examples

2.1.1 d-Hitting Set

d-HITTING SET
Input: A set U, a family $\mathcal{F} \subseteq 2^U$ such that $\forall_{a\in\mathcal{F}}|A| = d$ and an integer k.
Question: If there exists $X \subseteq U$ such that $|X| \leq k$ and $\forall_{A\in\mathcal{F}} A \cap X \neq \emptyset$?

We can observe that if we have $(k+1)$-sunflower $A_1, \ldots, A_{k+1} \in \mathcal{F}$ then if solution X exists then $X \cap \bigcap_{i=1}^{k+1} A_i \neq \emptyset$. In other case we need to take $k+1$ different elements from the sets of the sunflower to hit them all. So we can add $\bigcap_{i=1}^{k+1} A_i$ to the family $\mathcal{F}$, but then we do not need sets $A_1, \ldots, A_{k+1}$ there. Using that we can take $\mathcal{F}' = (\mathcal{F} \setminus \{A_1, \ldots, A_{k+1}\}) \cup \{\bigcap_{i=1}^{k+1} A_i\}$. And after applying such this rule we get $|\mathcal{F}| \leq d! \cdot k^d$ and therefore $|U| \leq d \cdot d! \cdot k^d$.

Alternatively we can say that if $|\mathcal{F}| > d! \cdot (k+1)^d$ then we have a $(k+2)$-sunflower. If X of size $\leq k$ hits some $(k+1)$ sets of this sunflower then it has to hit the intersection, and therefore X hits all $(k+2)$ sets of the sunflower. So we can throw away one of those sets from $\mathcal{F}$.

2.1.2 Triangle Hitting

TRIANGLE HITTING
Input: A graph G and an integer k.
Question: If there exists $X \subseteq V(G)$, such that $|X| \leq k$ and $G \setminus X$ is triangle-free?

We can view this problem as a special case of 3-Hitting Set. Let's declare a family of sets $\mathcal{F} := \{\{u, v, w\} | G[u, v, w]$ is a triangle $\}$. We will use alternative way of reduction for d-Hitting Set: We can take $(k+2)$-sunflower (if $|\mathcal{F}| > 3! \cdot (k+1)^3$) and throw out one of the sets of sunflower from $\mathcal{F}$. Then we throw out vertices which are in non of the sets of $\mathcal{F}$. Because we have still a $(k+1)$-sunflower then we still have to take something from the intersection, and then we would hit also our deleted set. After that we have at most $18(k+2)^3$ vertices in the graph. So we have a kernel for this problem of size $\mathcal{O}(k^3)$.

2.1.3 Dominating Set

DOMINATING SET
Input: A graph G and an integer k.
Question: If there exists $X \subseteq V(G)$, such that $|X| \leq k$ and $X \cup N(X) = V(G)$?

In the winter term we have shown that this problem is not FPT. Therefore there is no kernel.

Let's introduce some modifications of this problem.

Definition 7. A graph G is a *d-degenerate* if $\forall_{H\leq G} \exists_{v\in V(H)} deg_H(v) \leq d$.

d-DEG-DOM-SET
Input: A graph G and an integer k. Moreover a graph G is a d-degenerate
Question: If there exists $X \subseteq V(G)$, such that $|X| \leq k$ and $X \cup N(X) = V$?

RED-BLUE-DOM-SET
Input: A bipartite graph $G(R, B; E)$ and an integer k.
Question: If there exists $X \subseteq R$, such that $|X| \leq k$ and $N(X) = B$?

d-DEG-RED-BLUE-DOM-SET
Input: A bipartite graph $G(R, B; E)$ and an integer k. Moreover G is a d-degenerate.
Question: If there exists $X \subseteq R$, such that $|X| \leq k$ and $N(X) = B$?

We will study the last one. It is easy to reduce d-DEG-DOM-SET to $(d+1)$-DEG-RBDS and d-DEG-RBDS to $(d+1)$-DEG-DOM-SET.

As a first rule we can take:

(1) if we have $v, w \in R$ and $N(v) \subseteq N(w)$ then remove v.

Lets divide R into R_D and R_M that $v \in R_D \iff deg(v) > 2d$. We can observe that if (1) does not work then $|R_M| \leq \sum_{i=1}^{2d} \binom{|B|}{i}$, because neighbourhoods are pairwise different in R. So we have $|R_M| \leq \mathcal{O}\left(|B|^{2d}\right)$.

Let's define $H = G[R_D \cup B]$ and divide B into B_D and B_M that $v \in B_M \iff deg(v) \leq 4d$. In d-degenerate we have $|E| < d|V|$. Because H is also a d-degenerate then average degree is $< 2d$ and $|B_M| \geq \frac{|B|}{2}$.

If we have $|B| > 2 \cdot (4k)!(2dk + k + 1)^{4d}$ then $|B_M| > (4d)!(2dk + k + 1)^{4d}$. So if we define for $v \in B_M$ a set $A_v = N(v) \cap R_D$ then $|A_v| \leq 4d$ and from the sunflower lemma we have a $(2dk + k + 2)$-sunflower given by some vertices $v_1, \ldots, v_s \in B_M$.

But if there is a solution X then $X \cup R_M$ dominates $\leq 2dk$ vertices. So we have to dominate some $k + 2$ vertices of $v_1, \ldots, v_s$ by $R_D \cap X$. We can throw away v_s from the graph, because then any solution need to dominate $k + 1$ vertices of $v_1, v_2, \ldots, v_{s-1}$ and then still need to use some vertex of the intersection of sets of the sunflower. So v_s still would be dominated by this solution.

We can repeat this procedure while $|B| > 2 \cdot (4d)! \cdot (2dk+k+1)^{4d}$. After that we have $|B| \leq \mathcal{O}(k^{4d})$ and $|R_M| = \mathcal{O}(k^{\mathcal{O}(d^2)})$. Let n be the total number of vertices. The average vertex degree is $< 2d$ so we know that $\frac{|R_D| \cdot (2d+1)}{n} < 2d$. Therefore $n \leq (2d + 1)(|B| + |R_M|)$.

The first proof that d-DEG-DOM-SET has a polynomial kernel for fixed d is due to [7]. The proof above is from [2].

References

[1] Jonathan F. Buss and Judy Goldsmith. Nondeterminism within P. *SIAM J. Comput.*, 22(3):560–572, 1993.

[2] Marek Cygan, Fabrizio Grandoni, and Danny Hermelin. Tight kernel bounds for problems on graphs with small degeneracy, 2012. Manuscript.

[3] Marek Cygan, Stefan Kratsch, Marcin Pilipczuk, Michal Pilipczuk, and Magnus Wahlström. Clique cover and graph separation: New incompressibility results. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *ICALP (1)*, volume 7391 of *Lecture Notes in Computer Science*, pages 254–265. Springer, 2012.

[4] Marek Cygan, Marcin Pilipczuk, and Michal Pilipczuk. Known algorithms for edge clique cover are probably optimal. In *SODA*, pages 1044–1053, 2013.

[5] Holger Dell and Dieter van Melkebeek. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. In Leonard J. Schulman, editor, *STOC*, pages 251–260. ACM, 2010.

[6] Jens Gramm, Jiong Guo, Falk Hüffner, and Rolf Niedermeier. Data reduction and exact algorithms for clique cover. *ACM Journal of Experimental Algorithmics*, 13, 2008.

[7] Geevarghese Philip, Venkatesh Raman, and Somnath Sikdar. Polynomial kernels for dominating set in graphs of bounded degeneracy and beyond. *ACM Transactions on Algorithms*, 9(1):11, 2012.