RWTH Aachen **WS 2011/12**
Lehrgebiet Theoretische Informatik Problem Set 1
Langer—Reidl—Rossmanith—Sikdar 21.10.2011

Parameterized Algorithms Tutorial

Tutorial Exercise T1

A question from the lecture concerns the following algorithm for the Minimum Vertex Cover problem: choose an arbitrary edge $e = \{u, v\}$ that has not yet been covered and branch on the two subcases; on one branch include u in the solution and in the other include v in the solution. Return the smaller of the two solutions.

Does this algorithm run in FPT-time if parameterized by the size of the minimal vertex cover? If yes, provide a formal proof. If no, provide a generic counterexample.

Proposed Solution

This algorithm does not run in FPT-time. Consider the star graph S_r. The recursion tree of this algorithm on this graph will be one long path of length r with a single further vertex attach to each vertex of the path, in total this tree has $r + 1$ leaves. Now consider a graph consisting of l distjoint copies of S_r labeled $S_r^1, S_r^2, \ldots, S_r^l$. Without loss of generality, we assume that the algorithm first branches on the edges of S_r^1, then on the edges of S_r^2 and so on.

This results in a recursion tree of size $(r + 1)^l$: regard the whole search tree for a single star as a node with $r + 1$ children, then the depth of this reduced tree is exactly l.

The graph clearly has a minimum vertex cover of size r, but the size of the search tree and therefore the running time depends also exponentially on l. It follows that this algorithm does not run in FPT-time.

Tutorial Exercise T2

Design a parameterized algorithm for the Independent Set problem on planar graphs. Why does your approach not work in the case of general graphs?

Proposed Solution

Given Euler's fromula $|V| - |E| + |F| = 2$ for planar graphs, where F denotes the set of *faces*, i.e. closed regions in a drawing of G, we can derive a bound on the average degree of vertices in G.

Assuming that G is simple and contains more than three vertices, every edge of G can be adjacent to at most two faces[1] and, conversily, every face touches at least three edges. It follows that $3|F| \leq 2|E|$ which enables us to bound the number of edges linearly in $|V|$:

$$
\begin{aligned}
& |V| - |E| + |F| = 2 \\
\Rightarrow\quad & |V| - |E| + \frac{2}{3}|E| \geq 2 \\
\Leftrightarrow\quad & 3|V| - 6 \geq |E|
\end{aligned}
$$

[1] Note that the outer border of the graph's drawing is also considered a face

For the average degree it follows that

$$d_{avg} = \frac{2|E|}{|V|} \leq \frac{2(3|V|-6)}{|V|} = 6\frac{|V|-2}{|V|} < 6$$

and thus it follows that a planar graph G with more than three vertices has at least one vertex of degree five.

Finally, note that removing vertices from a planar graph yields again a planar graph. Therefore we can always branch on a vertex of degree five, including one of the six vertices in the independent set. The running time of this algorithm is $O(6^k\, poly(n))$.

In general graphs this approach does not work because we cannot garuantee a vertex of small degree.

Tutorial Exercise T3

Use the algorithm presented in the lecture to solve the following instance of CENTER STRING. Recall that the parameter is the hamming distance which we fix to be $d = 2$.

$$\begin{aligned} s_1 &= bbbb \\ s_2 &= aabb \\ s_3 &= aaaa \\ s_4 &= abaa \end{aligned}$$

Homework H1

The CLUSTER VERTEX DELETION problem is defined as follows: given a graph $G = (V, E)$ and an integer parameter k, does there exist a set S of size at most k such that $G[V \setminus S]$ consists of a collection of disjoint cliques. The cliques are disjoint in the sense that they do not share vertices and/or edges and there is no edge with one endpoint in one clique and the other in a different clique. Design an algorithm that runs in FPT-time wrt k as parameter. [Hint: A graph is a cluster graph if and only if it does not contain a path with three vertices as an induced subgraph.]

Proposed solution

Our branching algorithm works as follows: we search in $O(n^3)$ time three vertices in the graph that induce a P_3. If no such triple exists our graph is a cluster graph. Otherwise branch on the three vertices, each time removing one of them from the graph and decreasing the parameter by one.

This algorithm runs in time $O(3^k\, poly(n))$.

Homework H2

An input to the HITTING SET problem consists of a finite universe U and a set family $\mathcal{F} \subseteq 2^U$. The question is to decide whether there exists a set $H \subseteq U$ of size at most k such that $H \cap S \neq \emptyset$ for all $S \in \mathcal{F}$, that is, the set H contains at least one element from each set of the family $\mathcal{F}$.

A variant of this problem, known as 3-HITTING SET, imposes the additional restriction that $|S| \leq 3$ for all $S \in \mathcal{F}$. Design an algorithm for this variant that runs in FPT-time where k is the parameter.

Proposed solution

Similar to H1, we search choose some set and branch on the three possibilities of including one of the three elements in the hitting set. Further, we remove all elements from the sets that are already in the hitting set.

Each branch decreases the parameter by one, all in all the running time is again $O(3^k\, poly(n))$.