RWTH Aachen
Lehrgebiet Theoretische Informatik
Langer—Reidl—Rossmanith—Sikdar

WS 2011/12
Problem Set 8
18.12.2011

Parameterized Algorithms Tutorial

Tutorial Exercise T21

In this exercise we will use the technique of color-coding to design a randomized FPT-algorithm for the k-Path problem which is defined as follows: given a graph G and an integer k as input, decide whether G has a path on at least k vertices.

1. Show that if G does indeed have a path $\mathscr{P}$ on k vertices, then by assigning colors to the vertices of G from the set $\{1, \ldots, k\}$ uniformly at random, one colors the vertices of $\mathscr{P}$ with *distinct* colors with probability at least e^{-k}.

2. Now assume that we are given a graph whose vertices are colored using colors from the set $\{1, \ldots, k\}$. Show that one can find a path with k distinctly colored vertices in time $O(2^k \cdot k \cdot |E(G)|)$.

3. Use the above facts to design a randomized FPT-algorithm for the k-Path problem. What is the expected running time of the algorithm? Can you modify the algorithm for the k-Cycle problem?

Tutorial Exercise T22

We now consider the following idea of using only two colors to obtain a randomized FPT-algorithm for k-Path. As before assume that the input graph G has a path $\mathscr{P}$ on k vertices. Color the vertices of the graph using two colors, say, black and white uniformly at random. Show that the probability of obtaining a coloring where $\mathscr{P}$ is split into two roughly equal parts G_1 and G_2 is 2^{-k}. Use this idea and recursion to design an algorithm for k-Path. Analyze its running time.

Homework H17

The Triangle Packing problem is defined as follows: given a graph $G = (V, E)$ and an integer k, decide whether G has k vertex-disjoint 3-cycles. Use the idea of randomly coloring the vertices of G with k colors to enable easy detection of vertex-disjoint triangles. What is the expected running time of your algorithm?

Homework H18

The Partial Vertex Cover problem is defined as follows: given a graph G and integers k and t, decide whether there exists k vertices that cover at least t edges. The parameter is the integer t (when parameterized by k only, the problem is W[1]-complete). The point of this exercise is to use color-coding to obtain a randomized FPT-algorithm for this problem.

1. Show that if $t \leq k$ then the problem is polynomial-time solvable. What happens if the maximum degree of the input graph is at least t?

2. Now use the following idea for coloring the vertices of the graph with two colors, say, green and red. Assume that there exists $S \subseteq V(G)$ of size at most k such that S covers at least t edges. Color each vertex red or green with probability 1/2. Show that the probability that vertices in S are colored green and all vertices in $\{u \in V(G) \setminus S \mid (u, v) \in E(G) \text{ for some } v \in S\}$ are colored red is a function of k and t. Call such a coloring a *proper coloring*.

3. Given a properly colored graph, we now need to identify a solution quickly. Note that the green vertices decompose the graph into connected components and that these contain the potential solution vertices. Show that in a properly colored graph, the solution is always the union of some green components, that is, the solution either includes all vertices of a green component or none. Hence any green component with k or more vertices that does not cover at least t edges can be discarded. Use this to design an algorithm that identifies a solution set in a proper two-colored graph.

4. Use all the above facts to design a randomized FPT-algorithm for the problem and analyze its time complexity.