Homework 3

September 14, 2017

Definitions & Notation

Definition 1. *A* proper vertex coloring *of a graph G* using k colors *is a function $c : V(G) \to \{1, \ldots, k\}$ so that if $uv \in E(G)$, $c(u) \neq c(v)$.*

k-Coloring-VC	
Input:	A graph G, a vertex cover X of G with $\lvert X\rvert = \ell$, and a non-negative integer k
Parameter:	ℓ
Question:	Is there a proper vertex coloring of G using at most k colors?

Proofs Required

Theorem. k-Coloring-VC *is in FPT (there is an $f(\ell)n^{O(1)}$ algorithm).*

Proof. We prove that k-Coloring-VC is in FPT through the following steps. Let n be the number of vertices in the graph G.

1. First of all, we can remove those independent vertices in the graph G because we can fill any color since these vertices do not have any neighbors.

2. Decompose the vertex cover X and the rest of graph G, which is denoted by $G - X$. This step takes time linear in n, where n is the number of vertices of the graph G.

3. Then we check each vertex v in the subgraph $G - X$ that whether v has less than $k - 1$ neighbors that are vertices in the vertex cover X. Here we denote the set of neighbors, which are vertices in X, of the vertex v as N. The intuition of this step is the observation as follows.

   (a) If the vertex v has less than $k-1$ neighbors that are vertices in the vertex cover X, then the set $N \cup \{v\}$ must be k-colorable. Then we fill colors for each vertex in $N \cup \{v\}$.

   (b) If the vertex v has at least k neighbors that are vertices in the vertex cover X, we need to check that whether the set $N \cup \{v\}$ is k-colorable or not. If the set $N \cup \{v\}$ is not k-colorable, output $\perp$. If $N \cup \{v\}$ is k-colorable, then we fill the color for each vertex in $N \cup \{v\}$. There are $(\ell + 1)^k$ possibilities using brute-force method.

   Since each vertex in $G - X$ is connected to one or more vertices in X due to the definition of vertex cover and for every two vertices in $G - X$ they do not have connection to each over, the method above considers each vertex in the graph G and it is safe.

We focus on analyzing the (worst-case) running time of step 3. Since the size of $G-X$ is $n-\ell$ and the size of X is ℓ. Therefore, the worst-case running of the above algorithm is $(n-\ell)\cdot\ell\cdot(\ell+1)^k+O(n)$, since $k \leq \ell$, we have $(n-\ell)\cdot\ell\cdot\ell^k \leq (n-\ell)\cdot\ell\cdot(\ell+1)^\ell = (\ell+1)^\ell\cdot\ell\cdot n-(\ell+1)^\ell\cdot\ell^2 = O((\ell+1)^\ell\cdot\ell\cdot n)$, which is in FPT.

$\square$