Assignment 5 - Problem 7.18

Kousshik Raj (17CS30022)

14-10-2020

1 Problem 7.18

Show an algorithm that, given an n-vertex graph together with its tree decomposition of width at most k, solves ODD CYCLE TRANSVERSAL in time $3^k k^{O(1)} n$.

1.1 Solution

Let the given graph be $G = (V, E)$ and $(T, \{X_t\}_{t \in V(T)})$ be the given tree decomposition of G of width at most k. We assume that the given tree decomposition is a nice tree decomposition with root r, otherwise we can compute a nice tree decomposition from it in $O(k^2 \cdot \max(|V(T)|, |V(G)|))$ time (Lemma 7.4 in book). Now, we define T_t as the subtree rooted at node t and $G_t = G[\bigcup_{w \in V(T_w)} X_w]$. Define a coloring $C : X_t \to \{0, 1, 2\}$. That is, we color the vertices belonging to a bag with 3 different colors. Let S_t be a minimum odd cycle transversal of graph G_t. It can be seen that $G_t - S_t$ is a bipartite graph. The different colors indicate different scenarios.

- **Color 0:-** All these vertices occur in S_t, i.e, $C^{-1}(0) = X_t \cap S_t$.
- **Color 1:-** These vertices belong to partition A of the bipartite graph $G_t - S_t$.
- **Color 2:-** These vertices belong to partition B of the bipartite graph $G_t - S_t$.

We use $C_{|A}$ to denote coloring C restricted to the set A and $C_{v \to \alpha}$ to denote the coloring where color of vertex v is α. Now, let us look at a dynamic programming algorithm. Define $Z[t, C]$, where $t \in V(T)$ and $C : X_t \to \{0, 1, 2\}$, as the size of the minimum odd cycle transversal S_t, where $S_t \cap X_t = C^{-1}(0)$, $C^{-1}(1)$ and $C^{-1}(2)$ belong to partition A and B, respectively, of the bipartitie graph $G_t - S_t$. Thus, we can assume $C^{-1}(1)$ and $C^{-1}(2)$ are independent sets (if not, they cannot belong to the same partition), otherwise we can assign $Z[t, C] = \infty$. The transitions are,

Leaf Node: For a leaf node, $X_t = \phi$. Hence, $Z[t, \phi] = 0$.

Introduce Node: Let t be a node, whose child is t', that introduces vertex v. Then,

$$Z[t, C] = \begin{cases} 1 + Z[t', C_{|X_{t'}}] & C(v) = 0 \\ Z[t', C_{|X_{t'}}] & \text{otherwise} \end{cases}$$

In the case where $C(v) = 0$, v must be taken into the odd cycle transversal and we can use the previously calculated minimum size odd cycle transversal for the coloring that excludes v from its child. In the case where v belongs to one of the partition, we know that $C^{-1}(1)$ and $C^{-1}(2)$ are independent. And according to Lemma 7.3 (in book), the neighbours of v in G_t is a subset of $X_t \cup X_{t'}$ and thus cannot have neighbours else where. Thus, odd cycle transversal of the node is the same as the odd cycle transversal of the child.

Forget Node: Let t be a node, whose child is t', that forgets vertex v. Then,

$$Z[t, C] = \min(Z[t', C_{v \to 0}], Z[t', C_{v \to 1}], Z[t', C_{v \to 2}])$$

In this case, the only difference between the node and its child is vertex v and we try all possible coloring for that vertex in its child keeping the rest of the coloring same and choose the minimum among them all.

Join Node: Let t be a node, which joins t_1 and t_2. Then,

$$Z[t, C] = Z[t_1, C] + Z[t_2, C] - |C^{-1}(0)|$$

Since $X_t = X_{t_1} = X_{t_2}$, we need the minimum odd cycle transversal for the same coloring as the node in both of its children. And since we included the nodes in $C^{-1}(0)$ twice, we subtract it from the answer.

There are $|V(T)| \cdot 3^{(k+1)}$ states, and each transition can be done in $k^{O(1)}$ time. As $|V(T)| = O(kn)$ (Lemma 7.4 in book), our algorithm runs in $3^k k^{O(1)} n$.