CS 594 Graph Algorithms, Fall 2019, UIC
Homework 1
Instructor: Anastasios Sidiropoulos

Problem 1: Vertex Cover on graphs of low treewidth Let $G = (V, E)$ be a graph. Recall that a *vertex cover* in G is some $U \subseteq V$, such that for every $\{u, v\} \in E$, at least one of the two vertices u and v is in U (that is, $\{u, v\} \cap U \neq \emptyset$). The goal in the Vertex Cover is to find a vertex cover of minimum cardinality. Give an algorithm for solving the Vertex Cover problem on graphs of treewidth k, with running time $2^{O(k)}n^{O(1)}$.

Problem 2: An approximation scheme for Vertex Cover on planar graphs. We saw in class that Baker's technique can be used to obtain an approximation scheme for Vertex Cover on planar graphs. Show that the same method can be used to obtain an approximation scheme for Vertex Cover. Hint: Use the result from Problem 1.

Problem 3: The treewidth of a non-square grid. Recall that the treewidth of the $\ell \times \ell$ grid is $\Theta(\ell)$. Let n and k be integers, with $n \geq k$. Prove that the treewidth of the $n \times k$ grid is $\Theta(k)$.

Problem 4: Correlation Clustering. Suppose that you are given a set V of people, encoded by a complete undirected edge-weighted graph $G = (V, E)$. For each pair of people $x, y \in V$, if they like each other, then we set the weight of the edge $\{x, y\}$ to be 1; i.e. $w(\{u, v\}) = 1$. Otherwise, if x and y dislike each other, then we set $w(\{x, y\}) = -1$. We assume that there is no other possibility (i.e. a pair of people either mutually like or mutually dislike each other).

We wish to partition the set of people into two disjoint subsets so that we maximize the total number of pairs of people who like each other within each subset, plus the total number of pairs of people two disklike eachother across subsets. This can be formalized via the Correlation Clustering problem. An input consists of the complete graph $G = (V, E)$ and the weight function $w : E \to \{-1, 1\}$. The goal is to compute a bipartition $V = U_1 \cup U_2$, maximizing

$$\begin{aligned}\mathsf{happiness}(U_1, U_2) = &|\{\{x, y\} \in E : w(\{x, y\}) = -1, x \in U_1, y \in U_2\}| \\ &+ |\{\{x, y\} \in E : w(\{x, y\}) = 1, x \in U_1, y \in U_1\}| \\ &+ |\{\{x, y\} \in E : w(\{x, y\}) = 1, x \in U_2, y \in U_2\}|\end{aligned}$$

Design a polynomial-time approximation scheme for the above problem. That is, for any constant $\varepsilon > 0$, on input a graph with n vertices, your algorithm should have running time $n^{f(\varepsilon)}$, for some function $\varepsilon > 0$. Can you improve the running time to $g(\varepsilon) \cdot n^{O(1)}$, for some function g? Hint: Use the regularity lemma.

Problem 5: Feedback Vertex Set Let G be a graph of treewidth k. Prove that G has a balanced separator of size $O(k)$. That is, show that there exists some $X \subset V(G)$, with $|X| = O(k)$, such that every connected component of $G \setminus X$ has at most $\frac{9}{10}|V(G)|$ vertices.

Problem 6: From Sparse Cuts to Balanced Cuts Let G be a graph. Recall that the *sparsity* of a cut $(S, V(G) \setminus S)$ is defined to be

$$\varphi(S) = \frac{|E(S, V(G) \setminus S)|}{|S| \cdot |V(G) \setminus S|},$$

where $E(S, V(G) \setminus S)$ denotes the set of edges with exactly one endpoint in S.

Suppose that you are given (as a black box) a polynomial-time α-approximation algorithm $\mathcal{A}$ for the Sparsest-Cut problem. That is, on input a graph G, algorithm $\mathcal{A}$ outputs a cut $(S, V(G) \setminus S)$ with sparsity at most $\alpha \cdot \mathsf{OPT}$, where OPT is the minimum sparsity over all cuts in G. Using $\mathcal{A}_1$ as a black box, design a polynomial-time bi-criteria approximation algorithm $\mathcal{A}_2$ for the Balanced Cut problem. More precisely, on input a graph G, algorithm $\mathcal{A}_2$ outputs a cut $(Y, V(G) \setminus Y)$ satisfying the following conditions:

(1) $\frac{1}{10}|V(G)| \leq |Y| \leq \frac{9}{10}|V(G)|$.

(2) $|E(Y, V(G) \setminus Y)| \leq O(\alpha) \cdot \mathsf{OPT}'$, where OPT' is the minimum sparsity over all cuts that have exactly $\lceil |V(G)|/2 \rceil$ vertices on one side, and exactly $\lfloor |V(G)|/2 \rfloor$ vertices on the other side.