CR 13: Graph Decompositions
Homework 1

Date: 28/09/2021, to hand in by 26/10/2021
Total Marks: 20

- You can handwrite or type your composition.
- The 3 exercises are independent. You may tackle them in any order.
- For each question, you may of course consider the statements of the previous questions as true even if you could not prove them.
- We encourage you to draw figures whenever you feel that they will be useful for your reader.

1 Treewidth of a particular family of planar graphs (3 marks)

Let us consider the planar graph G_n on $3 \cdot 2^n - 2$ vertices obtained from two full binary trees with height n, hence 2^n leaves, by identifying pairs of homologous leaves and adding a path linking the identified leaves, in a planar way. See figure 1 for an illustration of G_3.

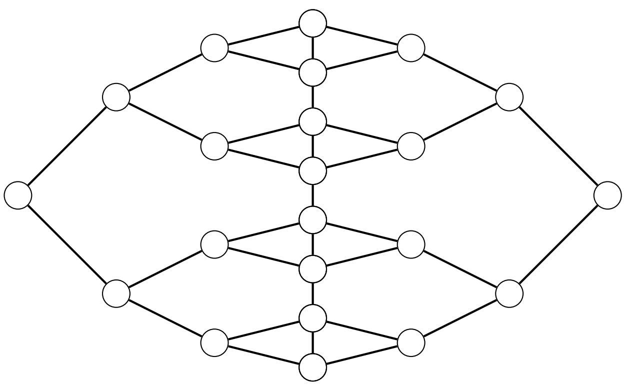

Figure 1: The planar graph G_3.

Q.1) Show that for every integer $n \geqslant 2$, the treewidth of G_n is at least 3 and at most 4. Bonus point, if you show that for every integer $n \geqslant 4$, the treewidth of G_n is precisely 4. 3+1 marks

2 Maximum k-Coverage in planar graphs (9 marks)

Maximum k-Coverage generalizes the k-Vertex Cover problem by asking whether k vertices touching at least p edges exist. Note that if one sets p to $|E(G)|$, Maximum k-Coverage is indeed equivalent to k-Vertex Cover. In particular, Maximum k-Coverage is NP-complete.

Maximum k-Coverage	**Parameter:** k
Input: A graph G and two positive integers k and p.	
Question: Is there a set $S \subseteq V(G)$ such that $\|S\| \leqslant k$ and at least p edges of G have at least one endpoint in S?	

Importantly we consider k as the parameter, and not p, nor a combination of p and k. Contrary to k-Vertex Cover, Maximum k-Coverage does *not* admit a fixed-parameter tractable (FPT) algorithm in general graphs, i.e., one with running time $f(k)|V(G)|^{O(1)}$ for some computable function f. The goal of this exercise is to design FPT algorithms for Maximum k-Coverage when restricted to planar graphs.

Q.2) Present an algorithm solving Maximum k-Coverage in time $2^t|V(G)|^{O(1)}$ when the (non-necessarily planar) input (G,k,p) comes with a nice tree decomposition of G of width t. Detail the correctness only in the case of the introduce node. 2.5 marks

Q.3) Using the previous question show that Maximum k-Coverage admits a $2^{O(k)}|V(G)|^{O(1)}$ time algorithm in planar graphs. 2 marks

We will now find a faster algorithm with running time $2^{O(\sqrt{k})}|V(G)|^{O(1)}$.

Q.4) Explain why the bidimensionality technique (small treewidth or large grid as minor or as edge contraction) does not work *as is* for the Maximum k-Coverage problem. 0.5 marks

We recall that $X \subseteq V(G)$ is a dominating set of G whenever $N[X] = V(G)$, that is, X and its neighborhood spans the entire vertex set of G.

Q.5) Given any graph G, find a polytime-computable ordering of its vertices, say, $v_1, v_2, \ldots, v_{|V(G)|}$, such that if the input (G,k,p) of Maximum k-Coverage has a solution, then it has one solution, S, such that there is an integer r satisfying both $S \subseteq \{v_1, \ldots, v_r\}$ and S is a dominating set of the graph $G[\{v_1, \ldots, v_r\}]$, i.e., the subgraph of G induced by $\{v_1, \ldots, v_r\}$.

Hint: the adequate ordering is not proper to planar graphs, and can break ties arbitrarily. 2 marks

For the next question, you can use without a proof that, there is a polytime algorithm that, given any planar graph G and positive integer k, outputs a nice tree decomposition of G of width $20k$ or an edge contraction of G isomorphic to the k-by-k triangulated grid.

Q.6) Deduce an algorithm solving Maximum k-Coverage in planar graphs in $2^{O(\sqrt{k})}|V(G)|^{O(1)}$. 2 marks

3 Breakable permutations (8 marks)

Let σ be a permutation of the set $[n] := \{1, \ldots, n\}$, that is a bijection of $[n]$ into itself. If S is a subset of $[n]$, we denote by $\sigma(S)$ the set of images of elements of S. Let $n \leqslant m$ be two positive integers. A permutation τ of $[n]$ is a *subpermutation* of σ of $[m]$ if there is an increasing injective function f from $[n]$ into $[m]$ such that for all pairs of distinct i, j in $[n]$ we have $\tau(i) < \tau(j)$ if and only if $\sigma(f(i)) < \sigma(f(j))$. If such an f exists then we write $\tau <_s \sigma$.

Q.7) Show that $<_s$ is a partial order. 0.5 marks

We say that σ of $[n]$ is *breakable* if $n = 1$ or if there exists $i \in [n-1]$ such that $\sigma([i]) = [i]$ or $\sigma([i]) = [n] \setminus [n-i]$, *and if this property also holds for all subpermutations of* σ.

Q.8) Find all permutations of [4] which are not breakable. 1.5 marks

Q.9) Propose an $O(n^2)$ algorithm which tests if a permutation of $[n]$ is breakable or not. 2 marks

Q.10) Show that $<_s$ is a well-quasi-order on the set of breakable permutations. 4 marks