Sparsity — tutorial 10

Applications of uniform quasi-wideness

Problem 1. Prove that if on a graph G Splitter wins the (ℓ, m, r)-Splitter game, then she also wins the $(\ell \cdot m, 1, r)$-Splitter game.

Problem 2. Prove that for a graph G, the least ℓ for which Splitter wins the $(\ell, 1, \infty)$-Splitter game on G is equal to the treedepth of G.

Problem 3. Prove that given a graph G, a subset of vertices Z, and $r \in \mathbb{N}$, the number $\mathrm{dom}_r(G, Z)$ can be computed in time $2^{|Z|} \cdot |Z|^{\mathcal{O}(1)} \cdot (n+m)$. How would it affect the running time of the algorithm for Distance-r Dominating Set presented during the lecture, if we used this procedure instead of brute-forcing through all the sets of k distance-r neighborhoods in Z?

Problem 4. Fix a nowhere dense class $\mathcal{C}$ and $r \in \mathbb{N}$. Prove that given $G \in \mathcal{C}$, $A \subseteq V(G)$, and $k \in \mathbb{N}$, one can compute in polynomial time an induced subgraph H of G and $B \subseteq A$ such that the size of H is bounded by a function of k, and in G there is an distance-r independent set contained in A if and only if in H there is an distance-r independent set contained in B.

Definition 0.1. Consider the following parameterized algorithm for the Distance-r Dominating Set problem: given G and k, is the a distance-r dominating set in G of size at most k. The algorithm iteratively constructs a sequence of *candidates* $D_1, D_2, D_3, \ldots$, each being a vertex set of size at most k, and a sequence of *witnesses* $w_1, w_2, w_3, \ldots$, each being a single vertex. Having constructed $D_1, \ldots, D_i$ and $w_1, \ldots, w_i$, the algorithm computes D_{i+1} and w_{i+1} as follows.

Candidate Step: Check whether there exists a set of size at most k that distance-r dominates $\{w_1, \ldots, w_i\}$. If no, then terminate the algorithm and conclude that G does not have a distance-r dominating set of size at most k. Otherwise, pick D_{i+1} to be any such set.

Witness Step: Check whether D_{i+1} is a distance-r dominating set of G. If yes, then terminate the algorithm returning D_{i+1}. Otherwise, pick w_{i+1} to be any vertex not dominated by D_{i+1} and proceed to the next round.

Problem 5. Show that if r is fixed and G is drawn from a fixed nowhere dense class $\mathcal{C}$, then the ith Candidate Step can be implemented in time $i^{\mathcal{O}(k)} \cdot (n+m)$, while every Witness Step can be implemented in time $\mathcal{O}(k(n+m))$.

Problem 6. Show that if r is fixed and G is drawn from a fixed nowhere dense class $\mathcal{C}$, then for $k = 1$ the algorithm terminates after a constant number of rounds.

Problem 7. Show that if r is fixed and G is drawn from a fixed nowhere dense class $\mathcal{C}$, then the algorithm terminates after $k^{\mathcal{O}(1)}$ rounds, and therefore can be implemented so that it runs in time $2^{\mathcal{O}(k \log k)} \cdot (n+m)$.

Problem 8. Suppose $r \in \mathbb{N}$, G is a graph, S is subset of vertices of G, and $(u_1, v_1), (u_2, v_2)$ are two pairs of vertices from G. We say that S *distance-r separates* (u_1, v_1) and (u_2, v_2) if every path of length at most r with one endpoint in $\{u_1, v_2\}$ and second in $\{u_2, v_2\}$ contains a vertex of S.

Prove that for every nowhere dense class $\mathcal{C}$ and integer $r \in \mathbb{N}$, there exist a constant $s_r \in \mathbb{N}$ and a function $N_r \colon \mathbb{N} \to \mathbb{N}$ such that the following holds. For every $m \in \mathbb{N}$, graph $G \in \mathcal{C}$, and set A of pairs of vertices of G with $|A| \geq N_r(m)$, there exist $S \subseteq V(G)$ and $B \subseteq A$ with $|S| \leq s_r$ and $|B| \geq m$ such that every pair of distinct pairs from B is distance-r separated by S.