Fine-grained Parameterized Algorithms

FPA · SoSe-2024 · tcs.uni-frankfurt.de/parameterized/ · 2024-07-03 · *2d88037*

Problem set 4: Treewidth and Dynamic Programming

Overview: With this problem set, you can train reasoning about graph decompositions and developping dynamic programming techniques for bounded-width graphs.

Instructions: For each skill, select **exactly one** problem below and submit your solution in Moodle; in your submission, make sure to repeat the problem that you are solving. The problems are roughly ordered by difficulty, choose a problem that you find non-trivial and interesting. (You are of course welcome to try the other problems as well and ask us for feedback.)

‼ **Skill-4a. Graph decompositions:** *I can mathematically reason about path and tree decompositions.* (For more context, see Chapter 7 in Cygan et al.)

easy **4.1 The clique is in the bag.** Let G be a graph with tree-decomposition $\mathcal{T} = (T, \{X_t\}_{t\in V(T)})$. Prove that every clique of G is contained in some bag X_t.

4.2 Becoming nice. Prove Lemma 7.4 from the lecture: Given a tree decomposition $\mathcal{T} = (T, \{X_t\}_{t\in V(T)})$ of a graph G of width at most k, one can in time $O(k^2 \cdot \max(|V(T)|, |V(G)|))$ compute a nice tree decomposition of G of width at most k that has at most $O(k|V(G)|)$ nodes.

4.3 Expander graphs have high treewidth. An n-vertex graph G is called an *α-edge-expander* if for every set $X \subseteq V(G)$ of size at most $n/2$ there are at least $\alpha \cdot |X|$ edges of G that have exactly one endpoint in X. Prove that the treewidth of an n-vertex d-regular α-edge-expander is $\Omega(n\alpha/d)$.

4.4 Pathwidth. Show that the pathwidth of an n-vertex tree is at most $\lceil \log n \rceil$. Construct a class of trees of pathwidth k and with $O(3^k)$ vertices.

4.5 Treedepth. A *rooted forest* is a union of pairwise disjoint rooted trees. The *depth* of a rooted forest is the maximum number of vertices in any leaf-to-root path. An *embedding* of a graph G into a rooted forest H is an injective function $f : V(G) \to V(H)$ such that, for every edge $uv \in E(G)$, the vertex $f(u)$ is a descendant of $f(v)$ or $f(v)$ is a descendant of $f(u)$. The *treedepth* of a graph G is equal to the minimum integer d, for which there exists a rooted forest H of depth d and an embedding of G into H.

Show that the pathwidth of a nonempty graph is always smaller than its treedepth.

!! <u>Skill-4b.</u> **Dynamic Programming:** *I can apply dynamic programming techniques for bounded-width graphs to design fixed-parameter tractable algorithms.* (For examples, see Section 7.3 in Cygan et al.)

4.6 Undirected Hamiltonicity. Prove that *Undirected Hamiltonicity*[1] is fixed-parameter tractable when parameterized by pathwidth. You may assume the existence of an algorithm $\mathbb{A}$ that, given an undirected graph G and a positive integer k, computes a (nice) path decomposition of width k of G or correctly decides that there is no such path decomposition. Furthermore $\mathbb{A}$ runs in time $f(k)n^{O(1)}$. (For this problem, you are required to directly construct a dynamic programming algorithm over the path decomposition of the input graph. It is also possible to invoke Courcelle's Theorem, but you should not do so here.)

4.7 Graph homomorphisms. A *homomorphism* from a graph H to a graph G is a function

$$\varphi : V(H) \to V(G),$$

such that for every edge $\{u, v\}$ of H it holds that $\{\varphi(u), \varphi(v)\}$ is an edge of G.

a) Prove that one can decide whether there exists a graph homomorphism from H to G in time

$$f(|H|) \cdot |V(G)|^{\mathsf{tw}(H)+O(1)}$$

by constructing a dynamic programming algorithm over the tree decomposition of H.

b) (optional, ☕☕☕) Prove that there is a bounded-error randomized algorithm to decide whether a graph H is a subgraph of a graph G in time

$$f(|H|) \cdot |V(G)|^{\mathsf{tw}(H)+O(1)}.$$

[1] That is, given an undirected graph G, decide whether G contains an Hamilton Cycle.