Classical Algorithms for Maximum Flow

R. Kyng & M. Probst Gutenberg

The exercises for this week will not count toward your grade, but you are highly encouraged to solve them all. This exercise sheet has exercises related to week 7. We encourage you to start early so you have time to go through everything.

To get feedback, you must hand in your solutions by 23:59 on April 25. Both hand-written and LaTeX solutions are acceptable, but we will only attempt to read legible text.

Exercise 1: Implementing Field Preservations for Cut-Link Tree Rotations

In Chapter 12, Section 12.3, we described the operations $\text{PL}\textsc{ink}(u, v)$ and $\text{PC}\textsc{ut}(u, v)$ by doing $O(\log n)$ tree rotations (in expectation). However, we omitted the details. Here we ask you to give the pseudo-code for the operation $\text{PT}\textsc{ree}\text{R}\textsc{otation}(v, w)$ where it is assumed that on input v is the parent of w and the operation manipulates the tree over the path such that v and w change position as described in the script. For simplicity, you are allowed to assume that w is the left child of v, and that the nodes $\mathit{left}_{\mathcal{P}}(v), \mathit{right}_{\mathcal{P}}(v), \mathit{left}_{\mathcal{P}}(w), \mathit{right}_{\mathcal{P}}(w)$ exist. Accompany your pseudo-code with a brief analysis that confirms that the run-time is indeed $O(1)$.

Exercise 2: Blocking Flows on Expander Graphs

Some classical algorithms for maximum flow have the unexpected behaviour that they often terminate much faster than their worst-case guarantee would suggest. One such algorithm is the blocking flow algorithm. The goal of this exercise is to show that expander graphs are one class of graphs for which blocking flow works particularly well.

In this exercise we develop an approximate s-t maximum flow algorithm for a ϕ-expander graph $G = (V, E)$ with n vertices and m edges. Each edge has unit capacity. Show the following.

1. Assume that the amount of s-t demand D is at most $1 - \epsilon$ of the maximum flow, i.e. the s-t min-cut is at least $1/(1 - \epsilon)D$. Show that $h = O(\frac{\log m}{\phi\epsilon})$ to send D flow from s to t in that case.