Colorado CSCI 5454: Algorithms
Exercises - 2020-09-17

Problem 1

Consider a variant of max flow where, in addition to edge capacities $c(u, v)$, we are given a **vertex capacity function** $k : V \to \mathbb{R}> 0$.

In addition to the usual constraints on a flow f, the total positive flow into a vertex must be at most its capacity:

$$\sum_{u \in V} \max \{f(u, v), 0\} \leq k(v).$$

This models, for example, a case where a router can only handle so many bits per second flowing into it (even if the connections between routers can handle more bandwidth).

Give an efficient algorithm for this problem. *Hint: reduce to traditional max flow.*

Problem 2

Part a

Give an efficent algorithm for this problem:

- **Input:** A directed, unweighted graph $G = (V, E)$; vertices $s, t \in V$.
- **Output:** The maximum number of paths from s to t that are *edge-disjoint*: no edge appears in multiple paths.

Hint: reduce to max flow; use Integrality Theorem.

Part b

Using your algorithm from part (a), solve this problem:

- **Input:** A directed, unweighted graph $G = (V, E)$; vertices $s, t \in V$.
- **Output:** The smallest set of edges so that, if we remove them, there is no longer any path from s to t.

Problem 3

Give an efficent algorithm for this problem:

- **Input:** A directed, unweighted graph $G = (V, E)$; vertices $s, t \in V$.
- **Output:** The maximum number of paths from s to t that are *vertex-disjoint*: no vertex (except for s and t) appears in multiple paths.

Hint: reduce to max flow; use Integrality Theorem.