CSCI3160: Regular Exercise Set 10

Prepared by Yufei Tao

Problem 1. Let $G = (V, E)$ be a weighted directed graph where the edge weights are given by the function $w : E \to \mathbb{Z}$; there are no negative cycles in G. Recall that Johnson's algorithm adds a vertex v_{dummy} to G, and computes the shortest path distance $spdist(v_{dummy}, v)$ from v_{dummy} to every vertex. Then, the weight of each edge (u, v) is modified to:

$$w'(u, v) = w(u, v) + spdist(v_{dummy}, u) - spdist(v_{dummy}, v).$$

Prove: $w'(u, v) \geq 0$.

Solution. The claim $w'(u, v) \geq 0$ is equivalent to $w(u, v) + spdist(v_{dummy}, u) \geq spdist(v_{dummy}, v)$. The latter inequality holds because $w(u, v) + spdist(v_{dummy}, u)$ gives the length of only *one* path from v_{dummy} to v, and therefore, is at least the shortest distance $spdist(v_{dummy}, v)$ from v_{dummy} to v.

Problem 2 (Textbook Exercise 24.1-3). Let $G = (V, E)$ a weighted directed graph that does not have negative cycles. Denote by s a vertex in V. Suppose that, for every vertex $v \in V$, there is a shortest path from s to v that has no more than L edges, where L is an integer at most $|V| - 1$. Design an algorithm to find the shortest paths from s to all the other vertices in $O(|E| \cdot L)$ time.

Solution. We proved in the lecture that, for any vertex $v \in V$, if there is a shortest path from s to v that has ℓ edges, Bellman-Ford's algorithm will have obtained the shortest-path distance from s to v after ℓ rounds of edge relaxations. This implies that, the $(L+1)$-th round of edge relaxations will not alter any shortest path distances. We can therefore terminate the algorithm after the $(L+1)$-th round. The total running time is $O(|E| \cdot L)$.

Problem 3 (Single *Sink* Shortest Paths). Let $G = (V, E)$ a weighted directed graph that does not have negative cycles. Denote by t a vertex in V. Design an algorithm to find the shortest path from every vertex $v \in V$ to t. Your algorithm must terminate in $O(|V||E|)$ time.

Solution. Reverse the directions of all the edges in G, and then apply Bellman-Ford's algorithm.

Problem 4 (Dynamic Programming Nature of Bellman-Ford's). Let $G = (V, E)$ a weighted directed graph that does not have negative cycles. Denote by s a vertex in V. If a path from s to some vertex $v \in V$ uses at most $\ell \in [0, |V| - 1]$ edges, we call it an *ℓ-path* from s to v. Given a vertex v and an integer $\ell \in [0, |V| - 1]$, define $spdist(s, v \mid \ell)$ as the smallest length of all the ℓ-paths from s to v. Prove: for $\ell \geq 1$, it holds that

$$spdist(s, v \mid \ell) \quad = \quad \min \begin{cases} spdist(s, v \mid \ell - 1) \\ \min_{u \in IN(v)} spdist(s, u \mid \ell - 1) + w(u, v) \end{cases} \tag{1}$$

where $IN(v)$ is the set of in-neighbors of v (namely, $u \in IN(v)$ if (u, v) is an edge in E).

Solution. It is obvious that the left hand side of (1) is at most the right hand side. Next, we will prove that the left hand side is at least the right hand side.

Let π be a shortest ℓ-path from s to v; the length of π is $spdist(s, v \mid \ell)$. If π has less than ℓ edges, then we have: $spdist(s, v \mid \ell) \geq spdist(s, v \mid \ell - 1) \geq$ the right hand side of (1). The rest of the proof will focus on the situation where π uses exactly ℓ edges.

Denote by p the predecessor of v on π (i.e., (p, v) is the last edge of π). Thus, the sequence of edges from s to p on π makes a shortest $(\ell - 1)$-path from s to p. It thus follows that

$$\begin{aligned} spdist(s, v \mid \ell) &= spdist(s, p \mid \ell - 1) + w(p, v) \\ &\geq \text{the right hand side of (1).} \end{aligned}$$