CSCI3160: Regular Exercise Set 10

Prepared by Yufei Tao

Problem 1. Let $G = (V, E)$ be a simple directed graph where each edge $(u, v) \in E$ carries a weight $w(u, v)$, which can be negative. Assume $V = \{1, 2, ..., n\}$ and G has no negative cycles. For any $u, v \in V$ and any integer $k \in [0, n]$, define $spdist(u, v| \le k)$ as the shortest length of any path from u to v where all the intermediate vertices are in $\{1, 2, ..., k\}$ (recall: an *intermediate vertex* is a vertex on the path other than the source and destination vertices). Prove:

$$spdist(u, v \mid\le k) = \min \begin{cases} spdist(u, k \mid\le k-1) \\ spdist(u, k \mid\le k-1) + spdist(k, v \mid\le k-1) \end{cases}$$

(Hint: First prove LHS $\le$ RHS and then prove LHS $\ge$ RHS.)

Problem 2. Let $G = (V, E)$ be a simple directed graph where each edge $(u, v) \in E$ carries a weight $w(u, v)$, which can be negative. Assume $V = \{1, 2, ..., n\}$. This time, G may or may not have a negative cycle. Given any $u, v \in V$ (where u and v can be identical), we define a *simple path* from u to v as a path π satisfying:

- π starts from u and ends at v;
- u is not an intermediate vertex of π;
- v is not an intermediate vertex of π;
- no intermediate vertex appears twice on π.

For any $u, v \in V$ and any integer $k \in [0, n]$, define $spdist(u, v| \le k)$ as the shortest length of any *simple* path from u to v where all the intermediate vertices are in $\{1, 2, ..., k\}$. Prove:

$$spdist(u, v \mid\le k) = \min \begin{cases} spdist(u, k \mid\le k-1) \\ spdist(u, k \mid\le k-1) + spdist(k, v \mid\le k-1) \end{cases}$$

(Hint: First prove LHS $\le$ RHS and then prove LHS $\ge$ RHS.)

Problem 3. Let $G = (V, E)$ be a simple directed graph where each edge $(u, v) \in E$ carries a weight $w(u, v)$, which can be negative. Assume $V = \{1, 2, ..., n\}$. G may or may not have a negative cycle. For any $u, v \in V$ and an integer $k \in [0, n]$, define $spdist(u, v| \le k)$ as the shortest length of any *simple* path (defined as in Problem 3) from u to v where all the intermediate vertices are in $\{1, 2, ..., k\}$. Prove: G has a negative cycle if and only if $spdist(u, u| \le n) < 0$ for some $u \in V$.

Remark. This implies that The Floyd-Warshall algorithm can detect the presence of negative cycles in $O(|V|^3)$ time.

Problem 4. Let $G = (V, E)$ be a simple directed graph where every edge (u, v) carries a weight $w(u, v)$, which can be negative. G has no negative cycles. 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: $w'(u, v) \geq 0$.

Problem 5. Let $G = (V, E)$ be a simple directed graph where every edge (u, v) carries a *non-negative* weight $w(u, v)$. Apply Johnson's algorithm to compute a new weight $w'(u, v)$ for each edge $(u, v) \in E$. Prove: $w'(u, v) = w(u, v)$.