Network Flow Algorithms: Exercise 1.4

Yuchong Pan

December 22, 2020

We re-define $d_k(j)$ to be the length of the shortest s-j path of length k, as in R. M. Karp's original paper [1].

Let $c' = c - \mu$. Let Γ_0 be a cycle of G. Then we have that

$$\begin{aligned} c'(\Gamma_0) &= \sum_{e \in E(\Gamma_0)} c'(e) = \sum_{e \in E(\Gamma_0)} (c(e) - \mu) = \sum_{e \in E(\Gamma_0)} c(e) - |\Gamma_0| \mu = c(\Gamma_0) - |\Gamma_0| \min_{\text{cycle } \Gamma \text{ of } G} \frac{c(\Gamma)}{|\Gamma|} \\ &\geq c(\Gamma_0) - |\Gamma_0| \cdot \frac{c(\Gamma_0)}{|\Gamma_0|} = c(\Gamma_0) - c(\Gamma_0) = 0. \end{aligned}$$

This shows that G with edge costs c' does not have negative-cost cycles. Hence, the Bellman-Ford algorithm correctly computes the shortest s-j paths for all $j \in V$. Let $d'_k(j)$ be the length of the shortest s-j path of length k with edge costs c'. By Exercise 1.2, there exists a simple shortest path P_j from s to any $j \in V$, which is of length $< n$. Hence, $c'(P_j) = \min_{0 \leq k \leq n-1} d'_k(j)$ and $c'(P_j) \leq d'_n(j)$ for all $j \in V$. This implies that $d'_n(j) \geq \min_{0 \leq k \leq n-1} d'_k(j)$ for all $j \in V$. On the other hand, let $\Gamma^* = \arg\min_{\text{cycle } \Gamma \text{ of } G} \frac{c(\Gamma)}{|G|}$. Then we have that

$$c'(\Gamma^*) = c(\Gamma^*) - |\Gamma^*| \min_{\text{cycle } \Gamma \text{ of } G} \frac{c(\Gamma)}{|\Gamma|} = c(\Gamma^*) - |\Gamma^*| \cdot \frac{c(\Gamma^*)}{|\Gamma^*|} = c(\Gamma^*) - c(\Gamma^*) = 0.$$

Let $v \in V(\Gamma^*)$. Let P be a simple shortest s-v path. Then $|P| < n$. Let $\ell \in \mathbb{N}$ be such that $|P| + \ell|\Gamma^*| \geq n$. Then the path P' formed by appending ℓ copies of Γ^* to the end of P is also a shortest s-v path. Hence, the subpath P'' of P' formed by the first n edges of P', which is an s-v^* path for some $v^* \in V$, is a shortest s-v^* path. Therefore, $d'_n(v^*) = c'(P'') = \min_{0 \leq k \leq n-1} d'_k(v^*)$. This proves that

$$\min_{j \in V} \max_{0 \leq k \leq n-1} \frac{d'_n(j) - d'_k(j)}{n-k} = 0.$$

Let $j \in V$. Let $0 \leq k \leq n$. Let P be a shortest s-j path of length k. Then $d_k(j) = c(P)$. It is clear that P is also a shortest s-j path of length k with edge costs c'. Hence, $d'_k(j) = c'(P)$. Since $d_k(j)$ is a path of length k, then we have that

$$d_k(j) = c(P) = \sum_{e \in E(P)} c(e) = \sum_{e \in E(P)} (c(e) - \mu + \mu) = \sum_{e \in E(P)} (c(e) - \mu) + |P|\mu$$

$$= \sum_{e \in E(P)} c'(e) + k\mu = c'(P) + k\mu = d'_k(j) + k\mu.$$

Hence, we have that

$$\begin{aligned}
\min_{j \in V} \max_{0 \le k \le n-1} \frac{d_n(j) - d_k(j)}{n-k} &= \min_{j \in V} \max_{0 \le k \le n-1} \frac{(d'_n(j) + n\mu) - (d'_k(j) + k\mu)}{n-k} \\
&= \min_{j \in V} \max_{0 \le k \le n-1} \frac{d'_n(j) - d'_k(j) + (n-k)\mu}{n-k} \\
&= \min_{j \in V} \max_{0 \le k \le n-1} \left(\frac{d'_n(j) - d'_k(j)}{n-k} + \mu \right) \\
&= \min_{j \in V} \max_{0 \le k \le n-1} \frac{d'_n(j) - d'_k(j)}{n-k} + \mu \\
&= 0 + \mu = \mu.
\end{aligned}$$

Next, we show that $d_k(j)$ can be computed by the following recurrence:

$$d_k(j) = \begin{cases} \min_{(i,j) \in E} \left(d_{k-1}(i) + c(i,j) \right), & k > 0, \\ 0, & k = 0, j = s, \\ \infty, & k = 0, j \neq s. \end{cases} \tag{1}$$

It is clear that $d_0(s) = 0$ and $d_0(j) = \infty$ for all $j \in V \setminus \{s\}$. Let $1 \le k \le n$. Let $j \in V$. Let P be a shortest s-j path of length k. Let (i^*, j) be the last edge of P. Then the subpath P' formed by all edges of P except (i^*, j) is a shortest s-i^* path of length $k-1$. Hence, $c(P') = d_{k-1}(i^*)$. This implies that

$$d_k(j) = c(P) = c(P') + c(i^*, j) = d_{k-1}(i^*) + c(i^*, j) \ge \min_{(i,j) \in E} \left(d_{k-1}(i) + c(i,j) \right).$$

For all $(i,j) \in E$, if P_i is a shortest s-i path of length $k-1$, then P_i appended by (i,j) is an s-j path, so $d_{k-1}(i) + c(i,j) = c(P_i) + c(i,j) \ge d_k(j)$. This implies that $\min_{(i,j) \in E}(d_{k-1}(i) + c(i,j)) \ge d_k(j)$. This proves (1). We give Algorithm 1 to compute μ and a cycle Γ such that $\mu = \frac{c(\Gamma)}{|\Gamma|}$. It is clear that the running time of Algorithm 1 is $O(nm)$. It remains to show that Γ^* returned by Algorithm 1 satisfies $\frac{c(\Gamma^*)}{|\Gamma^*|} = \mu$. We note that $p_k(j)$ stores a shortest s-j path of length k by following $p_k(j)$ backwards. Hence, P is a shortest s-j^* path of length n. This implies that P is not simple and hence contains at least one cycle. Since $\frac{d_n(j^*) - d_{k^*}(j^*)}{n-k^*} = \mu$, then we have that

$$\begin{aligned}
\frac{d'_n(j^*) - d'_{k^*}(j^*)}{n-k^*} &= \frac{(d_n(j^*) - n\mu) - (d_{k^*}(j^*) - k^*\mu)}{n-k^*} = \frac{d_n(j^*) - d_{k^*}(j^*) - (n-k^*)\mu}{n-k^*} \\
&= \frac{d_n(j^*) - d_{k^*}(j^*)}{n-k^*} - \mu = \mu - \mu = 0.
\end{aligned}$$

This implies that $d'_n(j^*) = d'_{k^*}(j^*) = \min_{0 \le k \le n-1} d'_k(j^*)$ is the length of the shortest s-j^* path. Hence, cycle Γ^* contained in P must have cost 0 with edge costs c'. Otherwise, we could have eliminated Γ^* to get a lower cost. We have that

$$\frac{c(\Gamma^*)}{|\Gamma^*|} = \frac{\sum_{e \in E(\Gamma^*)} c(e)}{|\Gamma^*|} = \frac{\sum_{e \in E(\Gamma^*)} (c(e) - \mu + \mu)}{|\Gamma^*|} = \frac{\sum_{e \in E(\Gamma^*)} (c(e) - \mu) + |\Gamma^*|\mu}{|\Gamma^*|}$$

$$= \frac{\sum_{e \in E(\Gamma^*)} c'(e)}{|\Gamma^*|} + \mu = \frac{c'(\Gamma^*)}{|\Gamma^*|} + \mu = \frac{0}{|\Gamma^*|} + \mu = 0 + \mu = \mu.$$

This completes the proof.

$d_k(j) \leftarrow \infty$ for all $0 \leq k \leq n, j \in V$
$d_0(s) \leftarrow 0$
$p_k(j) \leftarrow$ **null** for all $0 \leq k \leq n, j \in V$
for $k \leftarrow 1, \dots, n$ **do**
    for $(i, j) \in E$ **do**
        if $d_{k-1}(i) + c(i, j) < d_k(j)$ **then**
            $d_k(j) \leftarrow d_{k-1}(i) + c(i, j)$
            $p_k(j) \leftarrow i$
$\mu \leftarrow \infty$
for $j \in V$ **do**
    $\nu \leftarrow -\infty$
    for $k \leftarrow 0, \dots, n-1$ **do**
        $\nu \leftarrow \max(\nu, \frac{d_n(j) - d_k(j)}{n-j})$
    if $\nu < \mu$ **then**
        $\mu \leftarrow \nu$
        $j^* \leftarrow j$
$P = \{(v_1, v_2), \dots, (v_{n-1}, v_n)\} \leftarrow$ path formed by following p_n from j^* backwards
for $p \leftarrow 1, \dots, n-1$ **do**
    for $q \leftarrow p+1, \dots, n$ **do**
        if $v_p = v_q$ **then**
            return $\Gamma^* = \{(v_p, v_{p+1}), \dots, (v_{q-1}, v_q)\}$

Algorithm 1: An algorithm for computing the minimum mean-cost cycle.

References

[1] R. M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete mathematics*, 23(3):309–311, 1978.