CS499 Homework 10

Intersteller

Exercise 10.1

Since

$$\sum_{v\in V} f(s,v) = \sum_{v\in V\setminus S} f(s,v) + \sum_{v\in S} f(s,v)$$

we only need to prove that

$$\sum_{v\in S} f(s,v) = \sum_{u\in S-s,\ v\in V\setminus S} f(u,v)$$

Since

$$\sum_{v\in S} f(s,v) = -\sum_{v\in S-s} f(v,s)$$

we only need to prove that

$$\sum_{u\in S-s,\ v\in s+V\setminus S} f(u,v) = 0$$

It is obvious to see that

$$\sum_{u\in S-s,\ v\in S-s} f(u,v) = 0$$

So, we only need to prove that

$$\sum_{u\in S-s,\ v\in s+V\setminus S} f(u,v) + \sum_{u\in S-s,\ v\in S-s} f(u,v) = \sum_{u\in S-s,\ v\in V} f(u,v) = 0$$

According to the defination,

$$\sum_{u\in S-s,\ v\in V} f(u,v) = 0$$

Done.

Exercise 10.2

Define the minimum cut between i and j as minCut(i,j). According to the Max Flow Min Cut Theorem, minCut(s,r)≥k, minCut(r,t)≥k. Obviously, minCut(s,t)≥min{minCut(s,r),minCut(r,t)}≥k, which means there is a flow from s to r of value k.

Exercise 10.4

Suppose there is a s-t-path in G that has less that k edges. Then, at least one edge in the path moves more than one level forward, which contradicts Definition 10.3. So, $dist(s,t) \geq k$.

Exercise 10.5

Since dist$(s,t) = k$, we have a path $s \to u_1 \to u_2 \cdots u_{k-1} \to t$. For convenience, we call $s : u_0, t : u_k$. We construct $V_0, V_1 \cdots V_K$ as follows:

Step1:$u_0 \in V_0, u_1 \in V_1, u_2 \in V_2, \cdots, u_{k-1} \in V_{K-1}, u_k \in V_K$.

Step2:$\forall v \in V\backslash V_k$, if there is an edge (v,t), let V_{k-1} contain v. Obviously, $\forall n, 0 \leq n \leq k-2, u_n$ can't link to t. Otherwise, we can skip u_{k-1} and get a shorter path from s to t, which contradicts $dist(s,t) = k$.

Step3:$\forall v \in V\backslash(V_k \cup V_{k-1})$, if $\exists edge(v,w), w \in V_{k-1}$, let V_{k-2} contain v. $\forall n, 0 \leq n \leq k-3$, u_n can't link to any vertex in V_{k-1}.

$\vdots$

Step$k$:$\forall v \in V\backslash \bigcup_{i=2}^{k} V_i$, if $\exists edge(v,w), w \in V_2$, let V_1 contain v. Similarly, u_0 can't link to any vertex in V_2.

Step$k+1$:$\forall v \in V$ and $v \notin V_i, i \neq 1,2,3,\cdots,k, v \in V_0$.

Therefore, if $\mathrm{dist}(s,t) = k$, (G,s,t,c) has a k-layering.

Exercise 10.6

Obviously, (1)$s \in V_0$ and (2)$t \in V_k$ are satisfied, we consider condition (3). Since (G,s,t,c) is a flow network and $V_0, V_1, ..., V_k$ is an optimal layering, every edge in G moves at most one level forward and $dist_G(s,t) = k$. And we denote the path p as $s \to v_1 \to v_2 \to ... \to v_{k-1} \to t$. Then we can get $s \in V_0$, $v_1 \in V_1$, ... ,$v_i \in V_i$, ... ,$v_{k-1} \in V_{k-1}$ and $t \in V_k$. Each edge is between two adjacent layerings. Since residual network G_f only add a reverse edge of each edge in p, these additional edges are also between two adjacent layerings. So $V_0, V_1, ..., V_k$ satisfy condition (3) and it is a layering of (G_f, s, t, c_f).

Exercise 10.7

Because there is a path from s to t, we suppose $dist_G(s,t) = k$. Based on the construction method in 10.5, we can find a k-layering and it is an optimal layering. Therefore, every network(G,s,t,c) has an optimal layering, provided there is a path from s to t.

Exercise 10.8

We consider each while-loop of EK algorithm. In every iteration, EK algorithm choose p to be a shortest s-t-path in G_f. And we denote this feasible s-t-path p as $v_0 \to v_1 \to v_2 \to ... \to v_{k-1} \to v_k$ and $v_0 = s \in V_0$, $v_1 \in V_1$, ... ,$v_i \in V_i$, ... ,$v_{k-1} \in V_{k-1}$ and $v_k = t \in V_k$. Then, EK algorithm routes c_{min} flow along p. In G, $\exists (v_i, v_{i+1}) \in p$ where $i \in 0, 1, 2..., k$, $f(v_i, v_{i+1}) = c(v_i, v_{i+1}) = c_{min}$. In G_f, $c_f(v_i, v_{i+1}) = 0$ and $c_f(v_{i+1}, v_i) = c_{min}$.
If $V_0, V_1, ..., V_k$ is still an optimal layering, $c_f(v_i, v_{i+1})$ is always 0 because it is impossible for a feasible s-t-path p from v_{i+1} to v_i. Therefore, in every iteration, the total number of the edges from V_i to $V_{i+1}(\forall i \in 0,1,2,..,k-1)$ will minus at least one. Obviously, the number of these edges are less than or equal to m. After m iteration, if it is still optimal, the number of these edges are 0. There is no feasible s-t-path which $dist(s,t) = k$, and $dist(s,t)$ will be larger than k. So, after at most m iterations of the while-loop, $V_0, V_1, ..., V_k$ ceases to be an optimal layering.

Exercise 10.9

According to Exercise 10.8, a particular layering is no more optimal after at most m iterations. Since a layering is at least 1-layering and at most n-latering, after at most $m * n$ iterations, there is no optimal layering, which means there is no s-t-path, the algorithm terminates.

Exercise 10.10

According to Exercise 10.9, the Edmonds-Karp algorithm terminates after n $\cdot$ m iterations of the while-loop, which is to say, we can get the max flow f after finite steps by Edmonds-Karp algorithm.

Question

Can we use the Edmonds-Karp algorithm in multi-source network?