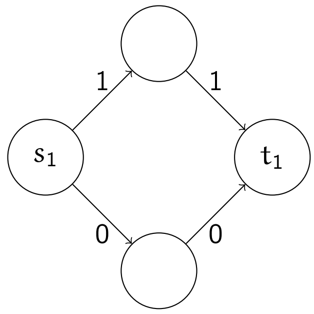
Homework 2

Mitar Milutinovic (24090156)*

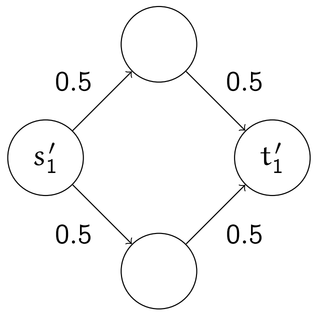
February 19, 2013

1.

1.(a)



Path routing from s_1 to t_1 has congestion equal to 1 (two possible paths are mirrored).



Fractional path routing from s_1' to t_1' has congestion equal to 0.5.

*Worked together with Daniel Aranki, Shiry Ginosar, Valkyrie Savage, Orianna De-Masi.

1.(b)

p_i is a set of all paths p_{i_j} that connect s_i and t_i. Length of path p_{i_j} is $d(p_{i_j})$. Each path p_{i_j} routes a fraction $f(p_{i_j})$ of common $f(p_i) = 1$, unit flow. $c(e)$ is congestion on edge e under routing, which is in our case sum of all fractional flows of all paths across edge e.

$$
\begin{aligned}
\max_e c(e) &\geqslant \sum_e d(e)c(e) \\
&= \sum_e d(e) \sum_i \sum_{j:p_{i_j} \ni e} f(p_{i_j}) \\
&= \sum_e \sum_i \sum_{j:p_{i_j} \ni e} f(p_{i_j}) d(e) \\
&= \sum_i \sum_j \sum_{e \in p_{i_j}} f(p_{i_j}) d(e) \\
&= \sum_i \sum_j f(p_{i_j}) \sum_{e \in p_{i_j}} d(e) \\
&= \sum_i \sum_j f(p_{i_j}) d(p_{i_j}) \\
&\geqslant \sum_i 1 \cdot \min_j d(p_{i_j}) \\
&= \sum_i d(s_i, t_i)
\end{aligned}
$$

3.

3.(a)

If after increasing the weight for δ on edge (u, v) cover is still feasible, we do not have to do anything. Otherwise, we start to repeatedly adjust the p. We maintain the priority queue Q of edges with the most infeasible edge at the front. We set $p(v) \leftarrow p(v) + \delta$ and add all other edges connected to v to Q. From Q we pop one edge (u', v') (the currently most infeasible edge) and compute new $\delta \leftarrow w(u', v') - p(u') - p(v')$. We set $p(u') \leftarrow p(u') - \delta$ and repeat the process of adjusting on u'. We repeat until δ becomes 0.

3.(b)

Algorithm si $O(m \log n)$, processing m edges in the graph in the priority queue order.