Ahuja 9.8 The primal-dual algorithm for min cost flows

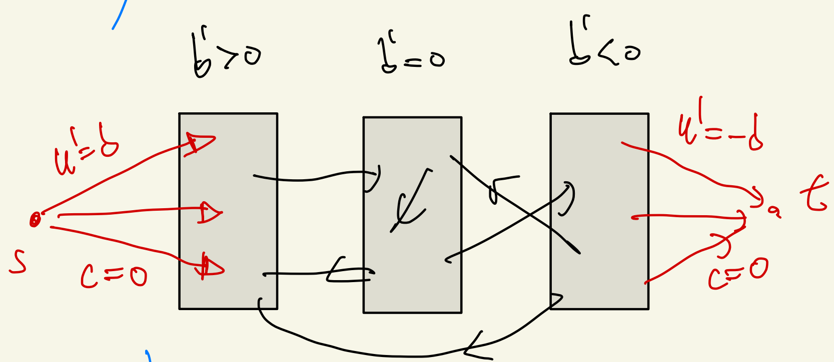
Recall from Ahuja 9.7 and BJG 3.10.2

Let $N' = (V', A', \ell' = 0, u', c', b')$ have $c'_{ij} \geq 0 \ \forall ij \in A$

and suppose $\exists$ a feasible flow in N.

Then we can find a min cost flow x as follows:

1) convert N' to $N = (V' \cup \{s,t\}, A' \cup A, \ell = 0, u, c)$



2) $x_{ij} \leftarrow 0 \quad \forall ij$

$\pi \leftarrow 0$

3) while $|x| < \sum_{b(v)>0} b(v) = K$

- Find shortest (s,t)-path P in $N(x)$ wrt cost c^{π}
- let $\tilde{x}$ be path flow of value $\delta(P)$ along P in $N(x)$
- $x \Leftarrow x \oplus \tilde{x}$

When the algorithm terminates it does so with a min cost flow.

In Ahuja this is shown by associating a potential $\pi : V \to \mathbb{R}$

- initially $\pi \equiv 0$ and $x \equiv 0$
- while $|x| < K$ do
  - let $d : V \to \mathbb{R}$ be shortest path dist from s in $N(x)$
  - $\pi \Leftarrow \pi - d$
  - $x \Leftarrow x \oplus \tilde{x}$ as above (augment along shortest path in $N(x)$)

Note that each arc ij on the shortest path P satisfies that $c_{ij}^{\pi} = 0$ wrt new π:

Before we update π we have

$$d(j) = d(i) + c_{ij}^{\pi} \quad \forall v_j \in P$$

$$\Updownarrow$$

$$c_{ij}^{\pi} + d(i) - d(j) = 0 \qquad (*)$$

$$\Updownarrow$$

$$c_{ij} - \pi(i) + \pi(j) - (-d(i)) + (-d(j)) = 0$$

so with new $\pi \leftarrow \pi - d$ we have

$$c_{ij}^{\pi - d} = c_{ij} - (\pi(i) - d(i)) + (\pi(j) - d(j))$$

$$= c_{ij} - \pi(i) + \pi(j) + d(i) - d(j)$$

$$= c_{ij}^{\pi} + d(i) - d(j) = 0 \text{ by } (*)$$

Hence we have $c_{ij}^{\pi} \geq 0$ for all arcs in $N(x)$ after updating x

$\Downarrow$ x is optimal by Theorem 9.3

New definition

- Given x, π let $N_0(x)$ be the subnetwork of $N(x)$ consisting of all vertices and those arcs ij for which $c_{ij}^{\pi} = 0$

- Every (s,t)-path in $N_0(x)$ is a shortest path

- If $\bar{x}$ is any (s,t)-flow in $N_0(x)$ then

  $x \oplus \bar{x}$ is optimal with value

  $|x| + |\bar{x}|$

New idea:

Instead of just sending flow along one (s,t)-path in $N_0(x)$ we find a maximum (s,t)-flow $\bar{x}$ is $N_0(x)$ and add it to x

By the remark above $x' = x \oplus \bar{x}$ is optimal and $|x'| > |x|$

If x' is a max flow (has value k) we are done

so assume this is not the case

What can we say about the distance from s to t in $N(x \oplus \bar{x})$?

Recall that $N(x \oplus \bar{x}) = N(x)(\bar{x})$ (BJG exercise 3.14)

so as $\bar{x}$ is a max flow in $N_0 \subseteq N(x)$ there is no (s,t)-path in $N_0(x \oplus \bar{x})$ implying that every (s,t)-path in $N(x \oplus \bar{x})$ uses at least one arc with $c_{ij}^{\pi} > 0$

where π is the current potential

- Calculate new distances from s in $N(x \oplus \bar{x})$ and denote them by d
- Let $\pi \leftarrow \pi - d$

Then $c_{ij}^{\pi} \geq 0 \ \forall ij \in N(x \oplus \bar{x})$ and the new $N_0(x \oplus \bar{x})$ will contain all arcs on shortest (s,t)-paths implying that we can repeat the step above by finding a max flow in $N_0(x \oplus \bar{x})$

This shows that the algorithm will return an optimal (min cost) flow in N'

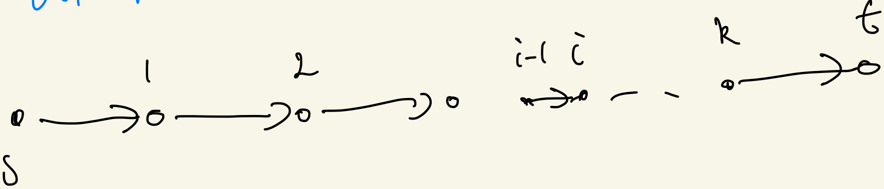
Complexity

- $\pi(s) = 0$ in the whole algorithm
- $\pi(t)$ decreases for each new maxflow calc in the current N_0 (as $d(t) > 0$ implies $\pi(t) - d(t) < \pi(t)$)

Claim $\pi(t) \geq -nC$: $C = \max\{|c_{ij}| \mid ij \in A\}$

The algorithm stops when there is no (s,t)-path in $N(x)$

look at the last iteration before x became max by $x \leftarrow x \oplus \bar{x}$ and let P be a shortest (s,t)-path in $N(x)$:



$$0 = c^{\pi}_{s1} = c_{s1} - \pi(s) + \pi(1) = c_{s1} + \pi(1)$$

$$\Downarrow$$

$-c_{s1} = \pi(1)$ so $\pi(1) \geq -C$

similarly for each i on P $0 = c^{\pi}_{i-1\,i} = c_{i-1\,i} - \pi(i-1) + \pi(i)$

$\Rightarrow$ $\pi(i) = \pi(i-1) - c_{i-1\,i} \geq \pi(i-1) - C$

induction $\Rightarrow$ $\pi(t) \geq -(n-1)C$

let $B = \max\{|b(v)| \mid v \in V\}$

At most $\min\{nC, nB\}$ iterations

(finding a max flow in N_0 and adding it)

$\Downarrow$ algorithm runs in time

$$O(\min\{nC, nB\} * MF) \text{ when}$$

MF is best complexity for max flow.