Lecture 20

Gidon Rosalki

date

1 Network flow 4 - Ford Fulkerson

1.0.1 Algorithm Ford-Fulkerson for optimal flow in a network

Input: a Normal network flow $N = (V, E, c, s, t)$
Output: An optimal flow f in N

1. **Preprocessing:** Extension: We will extend N to the extended network N'
2. **Initialisation:** $f' = 0$, $(\forall x, y \in V,\ f'(x, y) = 0)$
3. **Iteration:** We find the extended path P in the residual graph $G_{f'}$, and update $f' = f' + \Delta_{f',P}$
4. **Stop:** When there are no remaining extended paths paths in the residual graph $G_{f'}$, we stop
5. **Contraction:** We contract the extended flow f' to a normal flow f, and return f

1.0.2 Key theorems

Theorem 1 (1)**.** *We will assume that the input is a network of integer capacities. Then the algorithm stops after at most*

$$\sum_{x \in V \wedge (s,x) \in E} c(s, x)$$

iterations, and returns a flow with integer values.

Theorem 2 (2)**.** *If the FF algorithm stops, it returns an optimal flow.*

Theorem 3 (Lemma 1)**.** *Let N' be an extended network flow, and let f' be an extended flow in this network. Let P be the extended path in the residual graph $F_{f'}$, then $g = f' + \Delta_{f',P}$ is the extended flow in the network N', and additionally $|g| = |f'| + c_{f'}(P)$*

Proof . We will show that g is a correct extended flow:

1. Antisymmetry: According to the assumption f' enables antisymmetry, and according to the definition $\Delta_{f',P}$ enables antisymmetry. We will take $x, y \in V$:

$$\begin{aligned} g(x, y) &= f'(x, y) + \Delta_{f',P}(x, y) \\ &= -\left(f'(y, x) - \Delta_{f',P}(y, x)\right) \\ &= g(y, x) \end{aligned}$$

2. Conservation of mass: We will look at 2 types of node.
   (a) $x \notin p$: $\Delta_{f',p}(x, y) = 0$ for every $y \in V$, and from here

$$\sum_{y \in V} \Delta_{f',p}(x, y) = 0$$

(b) $x \in p$: Let there be y the node that comes before x in p, and z the node that comes after x.

$$\begin{aligned}\sum_{a\in V} \Delta_{f',p}(x,a) &= \sum_{a\in V\wedge a\neq y\wedge a\neq z} \Delta_{f',p}(x,a) + \Delta_{f',p}(x,z) + \Delta_{f',p}(x,y)\\ &= 0 + c_{f'}(p) + (-(c_{f'}(p)))\\ &= 0\end{aligned}$$

From here

$$\begin{aligned}\sum_{y\in V} g(x,y) &= \sum_{y\in V} (f'(x,y) + \Delta_{f',p}(x,y))\\ &= \sum_{y\in V} f'(x,y) + \sum_{y\in V} \Delta_{f',p}(x,y)\\ &= 0+0\end{aligned}$$

Where the first 0 comes from f' being an extended flow, and the second being due to what we've already shown.

3. Capacity constraint: $(x,y) \notin p$: We will note that $\Delta_{f',p}(x,y) \leq 0$, $g(x,y) = f'(x,y) + \Delta_{f',p}(x,y) \leq f'(x,y) \leq c;(x,y)$.
If $(x,y) \in p$:

$$\begin{aligned}\Delta_{f',p}(x,y) &= c_{f'}(p)\\ &= mnc_{f'}(e)\\ &\leq c_{f'}(x,y)\\ &= c'(x,y) - f'(x,y)\\ g(x,y) &= f'(x,y) + \Delta_{f',p}(x,y)\\ &\leq f'(x,y) + c'(x,y) - f'(x,y)\end{aligned}$$

so we have shown that g is an extended flow.
We will show that the flux increased by $c'(p)$. Let there be y, the node that comes after s in p.

$$\begin{aligned}|g| &= \sum_{i\in V} g(s,x)\\ &= \sum_{x\in V} (f'(s,x) + \Delta_{f',p}(s,x))\\ &= \sum_{x\in V} f'(s,x) + \sum_{x\in V} \Delta_{f',p}(s,x)\\ &= |f'| + \sum_{x\in V:x\neq y} \Delta_{f',p}(s,x) + \Delta_{f',p}(s,y)\\ &= |f'| + 0 + f_{f'}(p)\end{aligned}$$

□

Theorem 4 (Lemma 2)**.** *Under the assumption that the input to the FF algorithm is a flow with integer capacities, the extended flow in the extended network, for the length of the running of the algorithm is a flow with integer values.*

Proof . We will prove with induction on the iterations of the algorithm, that the flow is integer.
Basis: $f' = 0$ in initialisation $\implies$ the flow is integer
Iteration: We will assume that f' is an integer by definition, $\forall x, y \in V$ $c_{f'}(x,y) = c'(x,y) = f'(x,y)$, which is to say that the number is an integer according to the inductive assumption. From here for the extended path p, $c_{f'}(p)$ is an integer, and from here $\Delta_{f',p}$ is a flow with integer values, and therefore $f' + \Delta_{f',p}$ is also a flow in integers. □