Analysis of the Ford-Fulkerson Algorithm

Theorem 1. *Assume that all the capacities are integral then the run-time of the Ford-Fulkerson algorithm is $O(|E| \cdot |f|)$ where $|f|$ is the maximum flow value.*

Proof.

Steps 1-3 take $O(|E|)$ time.

The while loop is executed $O(|f|)$ times, since each iteration improves the flow by at least 1.

An augmenting path in the residual graph can be found by breadth-first or depth-first search, thus each iteration of the while loop takes $O(|E|)$ time.

□

The Edmonds-Karp Improvement

Theorem 2. *If the augmenting path in the Ford-Fulkerson algorithm is found by a breadth-first search (i.e. the algorithm uses a path of minimum number of edges), the run time of the algorithm is* $O(V \cdot E^2)$.

Let $\delta_f(s,u)$ be the minimum number of edges on a path from s to u is the residual graph G_f.

Lemma 1. *Let $f_1, f_2, ...$ be the sequence of flows produced by successive iterations of the Edmonds-Karp algorithm, then for any $u \in V$ the sequence $\delta_{f_i}(s,u)$ is non-decreasing in i.*

Proof.

Proof by contradiction. Let f' be the flow computed by adding the flow of an augmenting path in G_f.

$$B(f) \; = \; \{u \mid \delta_{f'}(s,u) < \delta_f(s,u)\}$$

and assume that $B(f)$ is not empty.

Let $v \in B(f)$ such that for all $u \in B(f)$

$$\delta_{f'}(s,v) \leq \delta_{f'}(s,u).$$

Let p' be a shortest path from s to v in $G_{f'}$. Let (u, v) be the last edge on that path.

Since $\delta_{f'}(s, u) < \delta_{f'}(s, v)$, $u \notin B(f)$.

Consider the flow from u to v in f.

If $f(u, v) < c(u, v)$ then

$$\begin{aligned}
\delta_f(s, v) &\leq \delta_f(s, u) + 1 \\
&\leq \delta_{f'}(s, u) + 1 \\
&= \delta_{f'}(s, v)
\end{aligned}$$

Thus, $v \notin B(f)$, contradiction.

If $f(u,v) = c(u,v)$ then $(u,v) \notin E_f$,

An augmenting path p in G_f that produces $G_{f'}$ must have the edge (v,u) so that $(u,v) \in E_{f'}$.

Thus, $\delta_f(s,u) = \delta_f(s,v) + 1$.

$$\begin{aligned}
\delta_f(s,v) &= \delta_f(s,u) - 1 \\
&\leq \delta_{f'}(s,u) - 1 \\
&= \delta_{f'}(s,v) - 2 \\
&< \delta_{f'}(s,v)
\end{aligned}$$

Contradicting $v \in B(f)$. $\square$

Theorem 3. *The total number of flow augmentations executed by the Edmonds-Karp algorithm is $O(VE)$.*

Proof.

An edge $(u, v) \in E_f$ is **critical** if

$$c_f(p) = c_f(u, v).$$

There is at least one critical edge in any augmenting path, and that edge does not appear in the next residual network.

Let $(u.v) \in E$, we bound the number of iterations in which (u, v) can be a critical edge.

Since the algorithm uses shortest path, if (u, v) is a critical edge in G_f,

$$\delta_f(s, v) = \delta_f(s, u) + 1$$

(u, v) can appear again in a residual network only after the flow on that edge is decreased, or there was augmentation on of flow on an augmented path that includes the edge (v, u).

Let $G_{f'}$ be the residual graph of this augmenting path. On that graph

$$\delta_{f'}(s,u) = \delta_{f'}(s,v) + 1$$

$$\begin{aligned}
\delta_{f'}(s,u) &= \delta_{f'}(s,v) + 1 \\
&\geq \delta_{f}(s,v) + 1 \\
&= \delta_{f}(s,u) + 2
\end{aligned}$$

Thus, every time the edge (u,v) becomes critical its distance to s (in number of edges) is increased by 2.

The distance cannot be more than $V - 2$, and there are $2E$ pairs of vertices that can be connected in the residual graph, thus there can be no more than $O(V \cdot E)$ augmentation iterations. $\square$

Corollary 1. *The run-time of the Edmonds-Karp algorithm is* $O(V \cdot E^2)$.

Bi-partite Matchning Through Maximum Flow

Given a bi-partite graph $G = (A, B, E)$ we can use a maximum flow algorithm to find maximum matching in G.

Define a directed graph $G' = (V'E')$:

- $V' = A \cup B \cup \{s, t\}$
- Connect s to all vertices in A. Direct all edges in E from A to B. Connect all vertices in B to t.
- All edges have capacity 1.

A flow is **integer value** if the flow through each edge is an integer value.

Theorem 4. *The Ford-Fulkerson algorithm generates an integer value flow in G'.*

Proof. By induction on the augmentation steps. □

Theorem 5. *The cardinality of the maximum matching in G equals the value of the maximum flow in G'.*

Proof. Since each node in A and B can be adjacent to only one vertex with integer flow > 0, a flow f in G' corresponds to a matching M in G with $|f| = |M|$. □