Ford-Fulkerson Algorithm

Adapted from the CLRS book slides

FORD-FULKERSON ALGORITHM

Keep augmenting flow along an augmenting path until there is no augmenting path.

Represent the flow attribute using the usual dot-notation, but on an edge: $(u, v).f$.

```
FORD-FULKERSON(G, s, t)
for each edge (u, v) ∈ G.E
    (u, v).f = 0
while there exists a path p from s to t in the residual network G_f
    c_f(p) = min {c_f(u, v) : (u, v) is in p}
    for each edge (u, v) in p
        if (u, v) ∈ G.E
            (u, v).f = (u, v).f + c_f(p)
        else (v, u).f = (v, u).f − c_f(p)
return f
```

FORD-FULKERSON ALGORITHM (continued)

Analysis

If capacities are all integer, then each augmenting path raises $|f|$ by ≥ 1. If max flow is f^*, then need $\leq |f^*|$ iterations $\Rightarrow$ time is $O(E\,|f^*|)$.

Note that this running time is *not* polynomial in input size. It depends on $|f^*|$, which is not a function of $|V|$ and $|E|$.

If capacities are rational, can scale them to integers.

If irrational, FORD-FULKERSON might never terminate!

EDMONDS-KARP ALGORITHM

Do FORD-FULKERSON, but compute augmenting paths by BFS of G_f. Augmenting paths are shortest paths $s \leadsto t$ in G_f, with all edge weights $= 1$.

Edmonds-Karp runs in $O(VE^2)$ time.

To prove, need to look at distances to vertices in G_f.

Let $\delta_f(u, v) =$ shortest path distance u to v in G_f, with unit edge weights.

Lemma

For all $v \in V - \{s, t\}$, $\delta_f(s, v)$ increases monotonically with each flow augmentation.

Theorem

Edmonds-Karp performs $O(VE)$ augmentations.

Use BFS to find each augmenting path in $O(E)$ time $\Rightarrow O(VE^2)$ time.

Can get better bounds.

MAXIMUM BIPARTITE MATCHING

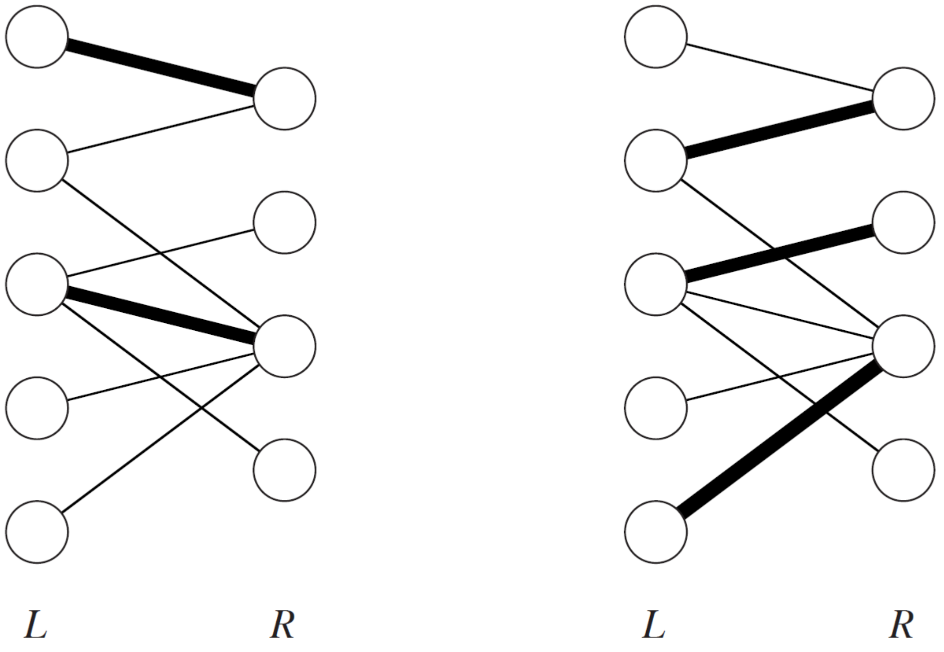
Example of a problem that can be solved by turning it into a flow problem.

$G = (V, E)$ (undirected) is ***bipartite*** if there is a partition of the vertices $V = L \cup R$ such that all edges in E go between L and R.

A ***matching*** is a subset of edges $M \subseteq E$ such that for all $v \in V$, ≤ 1 edge of M is incident on v. (Vertex v is ***matched*** if an edge of M is incident on it; otherwise ***unmatched***).

MAXIMUM BIPARTITE MATCHING (continued)

Maximum matching: a matching of maximum cardinality. (M is a maximum matching if $|M| \geq |M'|$ for all matchings M'.)



matching   maximum matching

MAXIMUM BIPARTITE MATCHING (continued)

Problem

Given a bipartite graph (with the partition), find a maximum matching.

Application

Matching planes to routes.

- L = set of planes.
- R = set of routes.
- $(u, v) \in E$ if plane u can fly route v.
- Want maximum # of routes to be served by planes.

MAXIMUM BIPARTITE MATCHING (continued)

Given G, define flow network $G' = (V', E')$.

- $V' = V \cup \{s, t\}$.
- $E' = \{(s, u) : u \in L\} \cup \{(u, v) : (u, v) \in E\} \cup \{(v, t) : v \in R\}$.
- $c(u, v) = 1$ for all $(u, v) \in E'$.